Universidad
Carlos III de Madrid

TRABAJO DE FIN DE GRADO

PREDICCIÓN DE ENERGÍA EÓLICA UTILIZANDO TÉCNICAS DE APRENDIZAJE AUTOMÁTICO

Autor: Malika Ouanani Allachi
Tutor: Inés María Galvan Leon
Tutor: Ricardo Aler Mur
U. CARLOS III DE MADRID

5-1-2015

Título: PREDICCIÓN DE LA ENERGÍA EÓLICA UTILIZANDO TÉCNICAS DE APRENDIZAJE AUTOMÁTICO
Autor: Malika Ouanani Allachi
Tutor: Inés María Galvan Leon
Tutor: Ricardo Aler Mur

EL TRIBUNAL:
Presidente: Carlos Ruiz Mora
Vocal: Cristina Martín Palacios
Secretario: Lorena Gonzáles Manzano
Suplente: Ginés Carrascal de las Heras

Realizado el acto de defensa y lectura del Trabajo Fin de Grado el día 5 de Marzo de 2015 en Colmenarejo, en la Escuela Politécnica Superior de la Universidad Carlos III de Madrid.

Resumen

El uso de energías renovables beneficia al medio ambiente, sustituyen la utilización de combustibles fósiles reduciendo de manera considerable el impacto medioambiental. La generación de energía eólica es una forma de generación no programable, ya que solo se produce energía cuando sopla el viento, que puede llegar a ser muy variable incluso en el corto plazo, por lo cual es difícil conocer con antelación y precisión suficiente la cantidad de energía eólica con la que podremos contar en cada momento. Esta variabilidad ocasiona más complejidad a su operación, por lo que su producción en el futuro tiene que ser estimada o prevista inevitablemente, para hacer factible el desarrollo e implantación de la energía eólica, y su integración en el sistema eléctrico.

Por ello, para este proyecto se ha tomado la decisión de estudiar y analizar diferentes métodos de regresión para la predicción de la energía eólica.

Para realizar este estudio y análisis se utiliza los datos de los siete parques eólicos de una competición de 2012. Con estos datos se estudian diferentes aproximaciones o modelos para abordar el problema de la competición, que consiste en la predicción de la generación de la energía eólica en un horizonte de 48 horas.

Para construir los diferentes modelos de predicción se ha utilizado la interfaz gráfica de la herramienta de análisis de datos Weka, y el lenguaje Java para dar formato y estructura al conjunto de datos de entrenamiento y test (asignar a las variables de entradas del modelo la variable de salida correspondiente), también se realiza una breve descripción de otras alternativas como herramientas y lenguajes de análisis y tratamiento de datos.

Para validar y evaluar los diferentes modelos utilizamos tres métodos de regresión, el método de regresión lineal, el Perceptron Multicapa y el M5P. De esta manera se aplican diferentes técnicas de aprendizaje automático tales como técnicas basadas en redes de neuronas, y árboles de regresión, para construir diferentes modelos lineales y no lineales de predicción.

En este trabajo se realiza tres alternativas de diseño de la solución, en otras palabras, tres alternativas para construir modelos de predicción. La primera alternativa consiste en generar modelos de predicción para la energía eólica basados en las variables meteorológicas que miden la velocidad y dirección del viento, la segunda alternativa consiste en generar modelos que incluyen variables de entrada con valores de instantes anteriores de la serie de la producción además de incluir el horizonte temporal (hora a la que se predice) y las variables meteorológicas utilizadas para el mejor modelo que resulte en la primera alternativa y por último la tercera alternativa consiste en generar modelos con diferentes bloques de horizontes de predicción.

Durante el estudio y análisis de cada modelo se realiza el estudio de los mejores parámetros y configuración para los métodos de regresión basados en redes de neuronas y árboles de regresión, se valida cada modelo de predicción para cada una de las granjas y finalmente se evalúa el mejor modelo con el test real o test de la competición.

Una vez estudiadas y analizadas las tres alternativas de diseño de modelos predictivos mediante diferentes técnicas de aprendizaje automático, se decide qué modelos y técnicas son más adecuados para realizar la predicción de la energía eólica para un horizonte de 48 horas, para cada una de las granjas con los datos proporcionados por una competición que se ha celebrado en el 2012.

Abstract

Renewable energy use benefits the environment, it subtitutes fossil fuels reducing considerably in this way the environmental impact. The production of wind energy isn't programmmable since it produces energy only when the wind blows, which can be highly variable even in the short term, so it is difficult to know in advance and sufficient precision the amount of wind energy that we can count with at all times. This variability brings more complexity to its operation, so its future production must inevitably be estimated or expected, to make possible the development and implementation of wind energy, and its integration into the electrical system.

Therefore, to carry out this project we decided to study and analyze different regression methods for predicting wind energy.

To perform this study and analysis, we used the data of seven wind farms of a competition in 2012. With these data we studied different approaches or models to address the problem of competition, which is the prediction of wind power generation in a horizon of 48 hours.

To construct the different prediction models we used the graphical interface tool Weka data analysis, and Java language to give format and structure to the training data and test set (To assign the model input variables to its corresponding output variables), we also perform a brief description of alternatives such as languages and tools of analysis and data processing.

To validate and evaluate the different models we used three methods of regression: linear regression method, the Multilayer Perceptron and M5P. Thus different automatic learning techniques such as techniques based on neuronal networks and regression trees are applied to construct different linear and nonlinear prediction models.

In this work, we have performed three alternatives for the solution design, in other words, three alternatives to build predictive models, the first one is to generate predictive models for wind power based on meteorological variables that measure the wind speed and direction , the second alternative is to generate models including input variables with values from previous moments of the series production as well as including the time horizon (time which is predicted) and meteorological variables used for the best model that results in the first alternative and finally the third alternative consists to generate models with different block prediction horizons.

During the study and analysis of each model, we perform the study of the best parameters and configuration for the regression methods based on neural networks and regression trees, we validate each prediction model for each of the farms and finally we evaluate the best model with the real test or the competition test.

Once the three design alternatives predictive models studied and analysed which uses different machine learning techniques, then we decided which models and techniques are more suitable for predicting wind power for a horizon of 48 hours for each of the farms with data provided by a competition that was held in 2012.

Tabla de contenido

Índice de Tablas.

Índice de Gráficos.

Capítulo 1. Introducción.

La energía eólica es la que utiliza la fuerza del viento para generar electricidad, para ello utiliza los aerogeneradores, que se encargan de mover una turbina y consiguen transformar la energía cinética del viento en energía mecánica.

La energía eólica es una fuente de energía barata, una energía limpia, no contaminante y es un tipo de energía renovable. La energía renovable se obtiene de recursos naturales y desechos, y puede ser energía eólica, solar, biomasa, hidráulica.

El uso de energías renovables beneficia al medio ambiente, sustituyen la utilización de combustibles fósiles reduciendo de manera considerable el impacto medioambiental, y si analizamos **el entorno socio económico** de las energías renovables vemos que conducen a otros beneficios como la creación de empleos locales y el desarrollo regional, lo que genera riqueza y empleo de calidad de manera directa e indirecta. Los sistemas de energía solar o eólica son capaces de suministrar energía a las regiones en vías de desarrollo o menos accesibles, las cuales pueden no tener los medios económicos o de infraestructura para utilizar combustibles fósiles, reduciendo así la dependencia energética de estas regiones respecto de terceros países.

La generación de energía eólica es una forma de generación no programable, ya que solo se produce energía cuando sopla el viento, que puede llegar a ser muy variable incluso en el corto plazo.

Para garantizar una integración adecuada de la generación de la energía eólica en los sistemas eléctricos se debe considerar la incertidumbre asociada a este tipo de fuente no gestionable.

Es difícil conocer con antelación y precisión suficiente la cantidad de energía eólica con la que podremos contar en cada momento. Esta variabilidad ocasiona más complejidad a su operación, por lo que su producción en el futuro tiene que ser estimada o prevista inevitablemente.

De esta manera la producción de la energía eólica se ve afectada por un error o incertidumbre de predicción. Si el viento disminuye, la potencia generada en los parques eólicos también disminuye, y esa falta de potencia debe ser reemplazada por otras fuentes de generación con una reserva suficiente en magnitud y velocidad de respuesta para que la demanda eléctrica no se vea afectada.

En otras ocasiones, puede ocurrir que no se pueda integrar en el sistema toda la producción eólica disponible, ya que la energía eólica no se genera de acuerdo a las necesidades de consumo, y sea necesario reducir el suministro de esta fuente de energía. Por todo esto, la predicción de generación eólica se ha convertido en un tema clave para hacer factible el desarrollo e implantación de la energía eólica, y su integración en el sistema eléctrico.

Por lo tanto resulta importante el problema de predicción de la energía eólica tanto para el operador del sistema como para los agentes del mercado o los propietarios de parques.

El operador del sistema eléctrico necesita conocer con antelación suficiente la cantidad de energía eólica que será inyectada en la red para gestionar la potencia que deberán generar las centrales convencionales, con el objetivo de cubrir la demanda total del sistema. Por otro lado, los agentes de mercado estarán interesados en conocer con la mayor certeza posible la potencia que generarán sus parques eólicos con el objetivo de seguir las estrategias que resulten más rentables en el mercado de energía eléctrica. Y por último, los propietarios de parques eólicos también estarán interesados en conocer en qué periodos se esperan menores potencias generadas en sus instalaciones, para afrontar labores programadas de mantenimiento.

La predicción de forma eficiente y efectiva de la energía eólica promueve a minimizar tanto los costes de operación del sistema como a maximizar los beneficios o minimizar las penalizaciones de los agentes de mercado.

La predicción eólica ayuda a avanzar hacia un mayor grado de penetración de la generación eólica en el sistema eléctrico, que sea compatible con la seguridad de operación y que permita integrar la mayor cantidad de energía renovable en el sistema.

Actualmente se utilizan varios modelos de predicción la producción de la energía eólica: modelos estadísticos lineales y no lineales de predicción, modelos físicos de meso-escala y de micro-escala, y modelos de curva de potencia [1].

Los avances en los modelos de predicción permiten alcanzar una mayor penetración de la energía eólica en el balance energético. Por otro lado conseguir una buena predicción eólica a corto plazo, (un plazo de 48 horas), lo suficiente para operar en el mercado de la electricidad, es fundamental y necesario para conseguir buenos resultados económicos.

El problema de predicción de la energía eólica a corto plazo se puede contemplar desde distintas escalas de tiempo, desde milisegundos a minutos, que se utiliza para el control activo del aerogenerador, hasta el intervalo de 48 a 72 horas, esta predicción es la que utiliza el explotador del parque eólico para realizar las ofertas en el mercado diario.

1.1. Objetivos

Partiendo de los datos sobre siete parques eólicos proporcionados en la siguiente página web https://www.kaggle.com/c/GEF2012-wind-forecasting, el objetivo principal de este trabajo es estudiar diferentes aproximaciones y modelos para abordar el problema de la competición [2], que consiste en la predicción de la generación de la energía eólica en un horizonte de 48 horas para cada uno de los siete parques.

El objetivo principal de este trabajo consiste en estudiar el comportamiento de diferentes métodos de aprendizaje automático, con la meta final de decidir el modelo y método de aprendizaje más adecuado, en el contexto de los datos de los que disponemos de la competición, que se pueden descargar de la página web arriba citada.

El trabajo se divide en tres partes:

Primera parte:

Se centra en estudiar y analizar los datos de los que disponemos, para decidir qué variables de las proporcionadas en la competición (U, V, WS, WD) son relevantes en la producción de la energía eléctrica final que se genera en cada granja eléctrica, y cuales simplemente resultan un ruido para el cálculo. Para cumplir con este propósito, se validan diferentes modelos utilizando la información disponible. El primero de ellos contendrá como información de entrada las cuatro variables de entradas (U, V, WS, WD), el segundo contendrá como información de entrada las dos primeras variables (U, V) y el tercer modelo contendrá las dos últimas variables (WS, WD).

A continuación procederemos a aplicar diferentes métodos y técnicas que la herramienta Weka nos proporciona.
En nuestro caso de estudio se centrará en utilizar tres métodos, la regresión lineal simple, el Perceptron Multicapa, y el algoritmo M5P. Una vez aplicados estos métodos sobre los tres modelos para cada granja eléctrica, analizaremos y compraremos los resultados obtenidos, para determinar de esta manera cuál es el método más eficiente para nuestro problema de predicción de la energía eólica.

Segunda parte:

Teniendo identificadas las variables disponibles (U, V, WS, WD) que influyen mejor en la solución del problema, incorporaremos como entradas valores anteriores de la serie temporal, la predicción de la energía eólica, para analizar y estudiar cómo influye en la predicción el uso de valores anteriores de la propia serie temporal para cada una de las siete granjas eléctricas. En este caso, también se realizará un estudio de diferentes modelos, lineal, Perceptron Multicapa, y el M5P, para determinar cuál es el método más eficiente para el problema de predicción de la energía eólica.

Tercera Parte:

Esta última parte consiste en construir modelos diferentes para grupos de horizontes temporales de predicción. Concretamente, se construyen tres modelos: el primer modelo abarca los horizontes desde 1 hora hasta 12 horas, el segundo modelo para los

horizontes desde 13 hasta 24 horas, y el último modelo para los horizontes de 25 hasta 48. Estos modelos se construyen, al igual que para las parte anteriores, con regresión lineal, PM y M5P. El objetivo es estudiar si la predicción para los 48 horizontes temporales puede ser mejorada utilizando modelos especializados en bloques de horizontes, en lugar de un modelo global para todos los horizontes.

1.2. Estructura de la memoria

En el capítulo 1 se presenta una breve introducción, donde se habla de la motivación, importancia de resolver el problema de predicción de la energía eólica y los diferentes modelos más utilizados actualmente, y por último se presentan los objetivos de este trabajo.

En el capítulo 2 se realiza una presentación de las diferentes alternativas de técnicas de aprendizaje automático y herramientas para trabajar con minería de datos, y se describe la herramienta de análisis de datos y métodos de regresión elegidos para llevar a cabo los diferentes experimentos realizados durante este trabajo.

En el capítulo 3 se realiza la descripción de los datos utilizados para abordar el problema de predicción de la energía eólica, así como los diferentes conjuntos de entrenamiento y test utilizados en las diferentes alternativas de diseño a la solución del problema de predicción, y por último se presenta el tipo de lenguaje de programación utilizado para tratar los datos que componen el conjunto de entrenamiento y test.

En el capítulo 4 se realiza la validación y evaluación de tres alternativas de diseño de modelos para predecir la energía eólica para un horizonte de 48 horas de predicción.

En el capítulo 5 se realiza una comparación entre los resultados obtenidos en las tres alternativas de análisis y estudio de una solución al problema de predicción de la energía eólica, estudiadas en el capítulo 4.

En el capítulo 6 se presenta la conclusión final de este trabajo y futuros trabajos.

En el capítulo 7 se presenta la planificación realizada para llevar a cabo este trabajo con éxito y el presupuesto que ocasiona realizarlo.

En el capítulo 8 se presentan las referencias consultadas para realizar este trabajo con éxito.

Capítulo 2. Herramienta y métodos de regresión utilizados

A continuación vamos a describir brevemente las características y ventajas de la herramienta Weka, que vamos a utilizar para realizar nuestros experimentos, y finalmente hablaremos de los métodos de regresión que nos ofrece Weka, y que utilizaremos para generar los diferentes modelos para resolver nuestro problema de predicción de la energía eólica.

2.1 La herramienta Weka.

Existen diferentes alternativas como herramientas de análisis y tratamiento de datos:

RapidMiner: Es una herramienta flexible para el análisis y la minería de datos. Proporciona una interfaz gráfica que simplifica las tareas complejas de minería de datos [3].

R: Es un proyecto de software libre, contribuye la posibilidad de cargar diferentes bibliotecas o paquetes con finalidades específicas de cálculo o gráfico [4].

Weka (Waikato Environment for Knowledge Analysis): Es un software libre distribuido bajo licencia GNU-GPL, para el aprendizaje automático y minería de datos escrito en Java y desarrollado en la Universidad de Waikato [5].

Para realizar el proyecto se ha utilizado la herramienta WEKA, porque es una herramienta con la que estoy familiarizada, fácil de usar y proporciona una gran variedad de algoritmos de aprendizaje automático mediante un interfaz común a todos ellos.

El paquete Weka contiene una colección de herramientas de visualización y algoritmos para análisis de datos y modelado predictivo, unidos a una interfaz gráfica de usuario para acceder fácilmente a sus funcionalidades.

Weka permite el pre-procesamiento de datos, clustering, clasificación, regresión, visualización, y selección de características. Todas las técnicas de Weka se basan en que los datos están disponibles en un fichero plano (*flat file*) o una relación, en la que cada registro de datos está descrito por un número fijo de atributos (normalmente numéricos o nominales). Estos atributos se corresponden con las variables de entrada y salida (en el caso de aprendizaje supervisado).

Las ventajas que ofrece la utilización de Weka son las siguientes:

- Está disponible libremente bajo la licencia pública general de GNU.
- Es muy portable porque está completamente implementado en Java y puede ejecutarse en casi cualquier plataforma.
- Contiene una extensa colección de técnicas para pre-procesamiento de datos y modelado.
- Es fácil de utilizar gracias a su interfaz gráfica de usuario.

2.2 Descripción de los métodos de predicción.

El problema de este caso de estudio es un problema de aprendizaje supervisado, con un conjunto de datos compuesto por patrones de entrada y la salida deseada, se trata de un problema de predicción o regresión debido a que la salida deseada es numérica.

Actualmente existen muchas alternativas como métodos de aprendizaje automático [9], sin embargo para resolver nuestro problema de predicción de la energía eólica se ha decidido utilizar dos técnicas de aprendizaje automático:

Las redes neuronales, el conocimiento se traduce en una serie de pesos y umbrales que poseen las neuronas. Concretamente utilizaremos el Perceptron multicapa. En un principio se iba a utilizar máquinas de vectores de soporte y finalmente se descartó porque requerían de alto tiempo de aprendizaje.

El aprendizaje de árboles, el conocimiento se transforma en un árbol de decisión o de regresión. Para nuestro caso hemos utilizado el M5P, basado en arboles de modelos.

Se ha decidido construir modelos lineales, no lineales y árboles, ya que previamente se desconoce el tipo de relación que existe entre las variables de entrada y las variables de salida (variable respuesta), se ha decidido construir diferentes modelos para comprobar con exactitud la relación existente entre las variables del problema, si es lineal, no lineal o jerárquica.

Para estudiar y analizar si la relación que existe entre las variables de entrada y salida es lineal. Se decidió utilizar el método de regresión lineal simple, ya que es el método más sencillo, simple y el más utilizado a la hora de predecir los valores de una variable cuantitativa (numérica) a partir de los valores de otra variable explicativa también cuantitativa.

Se ha decidido utilizar el Perceptrón Multicapa, para construir modelos no lineales, permitiendo de esta manera una perdida en la sencillez a costa de una mayor exactitud.

Se ha decidido construir modelos mediante arboles de regresión, ya que estos árboles permiten encontrar relaciones entre las variables de entrada y salida que tal vez no encontraríamos con los métodos más tradicionales. Los arboles de modelos de regresión permiten examinar los resultados y determinar visualmente como fluye el modelo.

Los árboles de predicción o regresión se utilizan cuando la clase a predecir es continua. En nuestro caso es posible aplicar arboles de modelos de regresión, dado que nuestro problema la clase es numérica o continua. El en árbol de regresión puro, cada nodo hoja almacena la media de las instancias que se clasifican con esa hoja. Si en las hojas se almacenan modelos lineales, se les denomina árboles de modelos de regresión.

A continuación describimos brevemente cada método y sus parámetros:

Regresión Lineal Simple: Es el esquema de aprendizaje para datos numéricos más sencillo, modela los datos usando una recta [6].

La **regresión lineal** o **ajuste lineal** es un método matemático que modela la relación entre una variable dependiente Y, las variables independientes Xi y un término aleatorio ε.

El modelo lineal simple:

$y = \beta 0 + \beta 1x + u$.

Elementos del modelo:

Y, x: Variables.

U: Término de error.

β1: Parámetro de pendiente en la relación entre x e y, es el cambio en y cuando se multiplica por el cambio en x. Es el parámetro clave en aplicaciones.

β0: Término constante (valor de y cuando x y u son cero). Menos interesante.

Relación funcional:

Si los demás factores contenidos en u se mantienen fijos, $\Delta u = 0$, entonces x tiene un efecto lineal sobre y $\Delta y = \beta 1 \Delta x$ si $\Delta u = 0$.

El nombre de la función de Weka que se utiliza para construir el modelo lineal simple es: SimpleLinearRegression.

Perceptrón Multicapa: Es un tipo de redes de neuronas artificial con aprendizaje supervisado que construye relaciones no lineales entre las variable de entrada y variable de salida [7].

Arquitectura de la red del Perceptrón Multicapa:

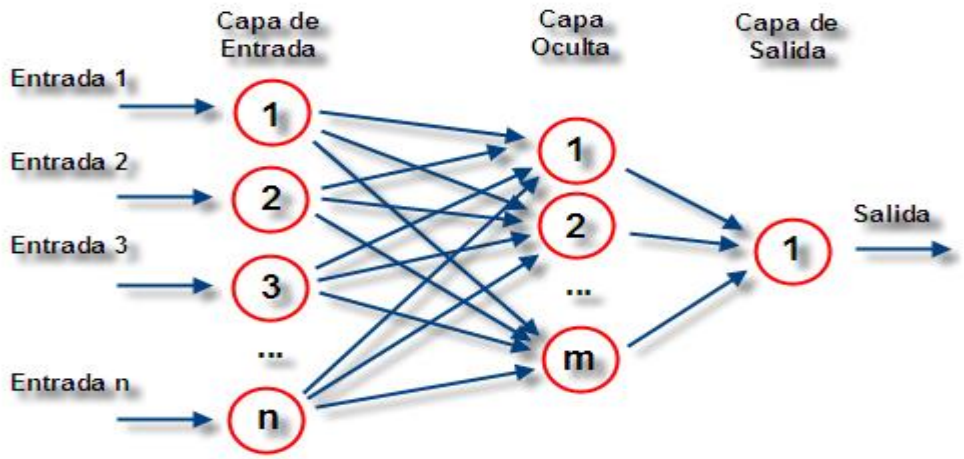
Está formada por múltiples capas, esto le permite resolver problemas que no son linealmente separables.

Capa de entrada: sólo se encargan de recibir las señales de entrada y propagarlas a la siguiente capa.

Capa de salida: proporciona al exterior la respuesta de la red para cada patrón de entrada.

Capas ocultas: Realizan un procesamiento no lineal de los datos recibidos.

Son redes "feedforward": conexiones hacia adelante. Generalmente cada neurona está conectada a todas las neuronas de la siguiente capa (conectividad total).



Arquitectura del Perceptron Multicapa.

Propagación de los patrones de entrada:

El Perceptrón Multicapa define una relación entre las variables de entrada y las variables de salida de la red. Esta relación se obtiene propagando hacia adelante los valores de las variables de entrada. Cada neurona de la red procesa la información recibida por sus entradas y produce una respuesta o activación que se propaga, a través de las conexiones correspondientes, hacia las neuronas de la siguiente capa.

Aprendizaje:

El aprendizaje del Perceptron Multicapa es un proceso iterativo supervisado: modificación paulatina de los parámetros de la red (pesos y umbrales) hasta que la salida de la red sea lo más próxima posible a la salida deseada o esperada para cada patrón de entrenamiento.

Dado el conjunto de patrones o ejemplos:

Vector de entrada: x(n)=(x1, x2,…, xn1)

Vector de salida deseada: s(n)

Encontrar Pesos W y umbrales U tales que: s(n) ≈ y(n) ∀ patrón n ⇔ |s(n)-y(n) | ≈ 0 ∀n ⇔ Minimizar E=Σ |s(n)-y(n) |

La función de Weka es MultilayerPerceptron, sus parámetros son los siguientes:

GUI: Proporciona una GUI para modificar o alterar la red neuronal durante el entrenamiento.

AutoBuild: Con esta opción los nodos de la red se generan automáticamente.

HiddenLayers: Este parámetro define el número de neuronas en cada capa oculta de la red neuronal. Si no se tienen capas ocultas el valor de este parámetro vale 0. En el caso de estar activa la opción autoBuild se tiene unos valores comodín para este atributo 'a' = (atributos + clases) / 2.

LearningRate: Tasa de aprendizaje. Este parámetro define cuanto de rápido queremos avanzar en la dirección opuesta al gradiente del error (dirección que minimice el error). En este caso se usará el valor por defecto de Weka.

Momentum: El momento permite a la red ignorar características pequeñas o de mínimo significado en la superficie del error. Sin el momento una red puede estancarse en un mínimo local, con momento la red puede deslizarse e evitar caer en un mínimo local. Este parámetro se usará su valor por defecto.

TrainingTime: Es el numero iteraciones con las cuales se va a entrenar la red neuronal. La red neuronal terminara cuando se hayan completado todas las iteraciones. En este caso se usará 1000.

ValidationSetSize: Tamaño en tanto por ciento del conjunto de validación. El entrenamiento de la red neuronal continuara hasta que se observe que el error en el conjunto de validación sea constantemente peor o se acaben el número de iteraciones del entrenamiento. Si el valor de este parámetro es 0 entonces no se utilizara un conjunto de validación por lo que el entrenamiento se terminara cuando se ejecute la última iteración. En este caso se usará 20.

ValidationThreshold: Es usado para determinar cuándo se termina la validación del test. Este valor dicta cuantas veces el error del conjunto de validación puede empeorar al ejecutar una iteración más. Se usará el valor por defecto.

MP5

El algoritmo MP5 construye árboles de modelos de regresión, los cuales contienen modelos lineales en los nodos hoja.

El algoritmo construye árboles de manera recursiva. En cada iteración elige un atributo para colocar en el nodo correspondiente y construye de manera recursiva los subárboles de cada una de las ramificaciones del nodo. Para definir el atributo con el que dividir, se emplea la varianza del error en cada hoja. Cuando se llega a los nodos hoja, se construye un modelo lineal con las instancias que hayan llegado a ese nodo.

Una vez construido el árbol que clasifica las instancias, se realiza la poda del mismo, tras lo cual, se obtiene para cada nodo hoja una constante en el caso de los árboles de regresión o un plano de regresión en el caso de árboles de modelos. En éste último caso, los atributos que formarán parte de la regresión serán aquellos que participaban en el subárbol que ha sido podado.

Al construir un árbol de modelos y definir, para cada hoja, un modelo lineal con los atributos del subárbol podado suele ser beneficioso, sobre todo cuando se tiene un pequeño conjunto de entrenamiento, realizar un proceso de suavizado que compense las discontinuidades que ocurren entre modelos lineales adyacentes. Este proceso consiste en que cuando se predice el valor de una instancia de test con el modelo lineal del nodo hoja correspondiente, este valor obtenido se filtra hacia atrás hasta el nodo hoja, suavizando dicho valor al combinarlo con el modelo lineal de cada nodo interior por el que pasa.

Para construir el árbol se emplea como heurística el minimizar la variación interna de los valores de la clase dentro de cada subconjunto. Se trata de seleccionar aquel atributo que maximice la reducción de la desviación estándar de error [8].

La función de M5 en Weka se denomina M5P y sus parámetros son los siguientes:

- **BuildRegressionTree:** Ya sea para generar un árbol de regresión o generar un conjunto de reglas en lugar de un árbol de modelos.

- **MinNumInstances:** Mínimo número de instancias para permitir que un nodo sea hoja.

- **SaveInstances** : Si desea guardar los datos de instancia en cada nodo del árbol

- **Unpruned**: Podar o no podar el árbol.

- **UseUnsmoothed**: Si se deben usar las predicciones sin filtrar.

Capítulo 3. Descripción de los datos.

Los datos que se utilizarán en este proyecto para abordar el problema de predicción de energía eólica proceden de una competición planteada en 2012 y fueron descargados de la siguiente página web: https://www.kaggle.com/c/GEF2012-wind-forecasting.

La información contenida en los ficheros de datos es de dos tipos. Por un lado, se proporcionan predicciones horarias de velocidad y dirección del viento, en siete granjas eólicas distintas. Por otro, se proporcionan series temporales horarias con la energía eléctrica producida en cada una de las siete granjas.

En total son diez ficheros de datos. Siete de estos ficheros, llamados "windforecasts_wf.csv", corresponden a cada una de las centrales o granjas eléctricas. Cada uno de estos ficheros contiene predicciones meteorológicas sobre el valor de la velocidad y dirección del viento a distintos horizontes temporales (desde la siguiente hora hasta las siguientes 48 horas), desde "2009-07-01" hasta "2012-06-26". La información sobre el viento se proporciona de dos maneras equivalentes, se presenta en forma vectorial (módulo y dirección: wind speed y wind direction) y en forma cartesiana (u: componente del vector velocidad en el eje x, v: componente del vector velocidad en el eje y).

Estos siete ficheros contienen datos con el siguiente formato "DATE, HORS, U, V, WS, WD", donde:

- DATE: corresponde con la fecha y hora en la que se realizó la predicción.
- HORS: Es la hora para la que se realiza la predicción, empieza en 1, y acaba en 48, se realiza la predicción durante 48 horas para cada hora determinada. Esto quiere decir que se realizan predicciones a 1 hora, a 2 horas, hasta 48 horas (2 días de antelación).
- U: Es el componente del vector velocidad en el eje x.
- V: Es el componente del vector velocidad en el eje y.
- WS: Es la velocidad del viento.
- WD: Es la dirección del viento.

Los tres ficheros restantes, llamados "train.csv", "test.csv" y "benchmark.csv" contienen las series temporales de energía eléctrica generada en el periodo de tiempo considerado. El primer fichero contiene los datos que se pueden usar para construir los modelos y que corresponden al periodo desde "01-07-2009" a las 00 horas hasta "26-06-2012" a las 12 horas. El segundo fichero (test) contiene una serie temporal de energía eléctrica para el periodo desde "01-01-2011" a la 1 horas hasta "28-06-2012" a las 12 horas, con huecos, para los que los modelos tienen que hacer la predicción y que sirve para evaluar los modelos con un conjunto de datos completamente independiente. El tercer fichero (benchmark) contiene el valor de la persistencia, que consiste básicamente en predecir la potencia en un instante de tiempo t con el valor de la potencia en el instante anterior, es decir WP(t)=WP(t-1). En este caso los datos no están divididos en varios ficheros para cada una de las siete granjas, sino que están todos contenidos en un único fichero. Tienen el siguiente formato: "DATE, WP1, WP2, WP3, WP4, WP5, WP6, WP7", donde date es el año seguido del mes seguido del día y seguido de la hora en el que se ha medido la energía eólica, y "WPX", es la producción

de energía en la granja X, donde x es el identificador de la central o granja eléctrica (se dispone de siete granjas).

Los diferentes ficheros disponibles poseen una estructura particular, dependiendo de que sean ficheros relativos a variables de entrada o variables de salida. Los siete ficheros con las predicciones meteorológicas para cada una de las granjas poseen la siguiente estructura: cada doce horas se dispone de la predicción del valor de las cuatro variables comentadas anteriormente, (U, V, WS, WD), para horizontes temporales de 1, 2, 3,…, hasta 48 horas, para cada central o granja eléctrica.

Ejemplo de fichero original de entrada correspondiente a la granja 1: windforecasts_wf1.csv

date,hors,u,v,ws,wd
2009070100,1,2.34,-0.79,2.47,108.68
2009070100,2,2.18,-0.99,2.4,114.31
2009070100,3,2.2,-1.21,2.51,118.71
…..

2009070100,46,2.75,-2.46,3.69,131.85
2009070100,47,2.3,-2.18,3.17,133.5
2009070100,48,1.93,-1.87,2.69,134.12
2009070112,1,2.77,-0.65,2.85,103.17
2009070112,2,3.12,-0.74,3.2,103.36
2009070112,3,3.29,-0.62,3.35,100.63
…..

2012062612,46,1.63,1.61,2.29,45.42
2012062612,47,1.07,1.58,1.91,34.05
2012062612,48,0.53,1.48,1.58,19.68

En cuanto al fichero “train.csv”, se observa que desde la fecha 2009/07/01 a las 00 horas, se anota el valor de la potencia o energía eléctrica generada en cada una de las siete centrales o granjas, hasta llegar a la fecha 01-01-2011 a las 00 horas. A partir de esta fecha, solo se anota el valor de la energía eléctrica pasadas 48 horas, en otras palabras, espera 48 horas para registrar el valor de la energía producida durante los 36 horas siguientes, así sucesivamente hasta llegar al final 2012/06/26 a las 12 horas.

A continuación se muestra un volcado del fichero de salida: train.csv

date,wp1,wp2,wp3,wp4,wp5,wp6,wp7
2009070100,0.045,0.233,0.494,0.105,0.056,0.118,0.051
2009070101,0.085,0.249,0.257,0.105,0.066,0.066,0.051
2009070102,0.02,0.175,0.178,0.033,0.015,0.026,0
2009070103,0.06,0.085,0.109,0.022,0.01,0.013,0
….

2010123123,0.677,0,0.968,0.722,0.88,0.882,0.833
2011010100,0.551,0,0.968,0.645,0.9,0.921,0.833

2011010301,0.371,0.28,0.316,0.865,0.581,0.908,0.859
2011010302,0.466,0.508,0.257,0.898,0.541,0.908,0.909
2011010303,0.466,0.693,0.267,0.871,0.46,0.895,0.909
….

2012062610,0.251,0.09,0.247,0.132,0.329,0.132,0.076
2012062611,0.301,0.058,0.356,0.061,0.293,0.105,0.076
2012062612,0.226,0.069,0.227,0.011,0.329,0.079,0

Y por último el fichero "test.csv", contiene el mismo formato que el fichero "train.csv", precedido por el "id", que es un identificador de cada línea registrada en el fichero. Este fichero contiene la fecha y el valor de la energía eléctrica generada en cada una de la siete granjas eléctricas, desde la fecha 2011/01/01/01h hasta el 2012/06/28/12h. En este fichero se anota la energía durante 48 horas seguidas, se espera 36 horas sin registrar nada, y se vuelve a registrar durante 48 horas, hasta el final del fichero. Esos huecos de 36 horas son los que se utilizan para evaluar los modelos construidos con los datos de entrenamiento.

A continuación se muestra un volcado del fichero de salida: test.csv

id,date,wp1,wp2,wp3,wp4,wp5,wp6,wp7
1,**2011010101**,0.471,0,0.968,0.733,0.92,0.763,0.808
2,2011010102,0.491,0,0.949,0.788,0.94,0.579,0.859
3,2011010103,0.551,0.196,0.968,0.485,0.956,0.553,0.682
….

7486,2012062810,0.085,0.09,0.296,0.22,0.02,0.184,0.202
7487,2012062811,0.09,0.021,0.247,0.303,0.005,0.224,0.177
7488,**2012062812**,0.095,0.032,0.188,0.193,0,0.132,0.076

3.1. Generación de los conjuntos de entrenamiento

Dada la estructura de los datos disponibles (descrita anteriormente), es necesario extraer los datos en el formato adecuado de patrones (entrada/salida) para entrenar los modelos de aprendizaje automático estudiados en este proyecto. Como ya se ha mencionado en la introducción, en este proyecto se estudian tres maneras diferentes de abordar la predicción de la energía eólica en un horizonte de predicción de 48 horas: utilizando un modelo global que se construya a partir de variables meteorológicas, utilizando un modelo global que incorpore valores anteriores de la serie de producción y construyendo tres modelos para tres bloques diferentes de horizontes de predicción. Este estudio requiere de la generación de diferentes conjuntos de entrenamiento para cada una de estas aproximaciones.

Para ello hemos definido funciones en el lenguaje de programación orientado a objetos Java, para generar los ficheros de entrenamiento finales con todas las estructuras necesarias, que utilizaremos para abordar el problema de predicción de energía eólica estudiado en este proyecto. Los ficheros generados poseen el formato

requerido por Weka (herramienta utilizada para construir los modelos) y extensión *arff*.

A continuación se describe el proceso para generar los diferentes conjuntos de entrenamiento para cada una de las aproximaciones estudiadas.

3.1.1. Conjunto de entrenamiento para la primera parte.

En primer lugar y utilizando los programas desarrollados en Java, se generan ficheros genéricos en formato Weka, de los cuales se extraerán las variables necesarias. Para ello se procesan los siete ficheros originales que contienen los valores de entradas (windforecasts_wf1.csv) que corresponden a cada una de las granjas eléctricas y se procesan también los ficheros (train.csv, test.csv) que contienen las salidas (”WP”). Se asignará a cada conjunto de valores de entradas el valor de la salida correspondiente, y se almacenará en el fichero correspondiente. En total se generarán siete ficheros con sus correspondientes valores de entradas y salida, cada uno de estos ficheros corresponde a una de las granjas eléctricas.

Ejemplo de fichero genérico en formato “.arff” con los valores de entradas y salida:

```
@relation entrenamiento
@attribute fecha numeric
@attribute mes numeric
@attribute dia numeric
@attribute hora numeric
@attribute horaPrediccion numeric
@attribute U numeric
@attribute V numeric
@attribute Ws numeric
@attribute Wd numeric
@attribute pw numeric
@data
2009,07,01,00,1,2.34,-0.79,2.47,108.68,0.085
2009,07,01,00,2,2.18,-0.99,2.4,114.31,0.02
2009,07,01,00,3,2.2,-1.21,2.51,118.71,0.06
…
2009,07,01,12,2,3.12,-0.74,3.2,103.36,0.01
2009,07,01,12,3,3.29,-0.62,3.35,100.63,0
2009,07,01,12,4,3.31,-0.37,3.33,96.42,0
….
```

Debido a que el objetivo es valorar en qué medida influyen las variables (U, V) y las variables (WS, WD) en la solución, para esta primera parte del trabajo se generan tres ficheros de entrenamiento para cada una de las granjas, a partir de los ficheros genéricos. De esta manera los ficheros resultantes que vamos a utilizar serían los siguientes:

- Ficheros con las cuatro variables como entrada de datos
- Ficheros con solo dos variables de entradas (U,V)

- Ficheros con las dos últimas variables (WS, WD).

Con la ayuda de la herramienta Weka y partiendo de los ficheros genéricos creados anteriormente, se eliminan los atributos no necesarios según cada una de las estructuras que se exponen a continuación y se guarda el fichero con su correspondiente identificativo o nombre para cada una de las granjas.

Generamos tres ficheros para cada granja según las siguientes estructuras:

Estructura 1: Fichero de entrenamiento con los atributos U, V, WS, WD, PW. Cada línea de este fichero contiene el valor horario correspondiente a las predicciones meteorológicas U, V, WS, WD y producción de la energía o potencia eléctrica PW, es decir, cada línea contiene el valor correspondiente a los atributos U, V, WS, WD y PW para cada fecha y hora, desde el periodo "2009-07-01-00h" hasta "2010-12-31-12h".

Ejemplo de fichero en formato ".arff" con la primera estructura:

@relation entrenamiento1-weka.filters.unsupervised.attribute.Remove-R1-5

@attribute U numeric
@attribute V numeric
@attribute Ws numeric
@attribute Wd numeric
@attribute pw numeric

@data
2.34,-0.79,2.47,108.68,0.085
2.18,-0.99,2.4,114.31,0.02
2.2,-1.21,2.51,118.71,0.06
….

Las primeras líneas contienen la información correspondiente a la cabecera para ficheros con la extensión ".arff" para poder usar la herramienta Weka, en la cabecera se declaran los atributos y después de "@data" se introduce el conjunto de datos de entrenamiento, cada línea contiene los valores correspondientes a los atributos declarados anteriormente separados por coma ",".

Estructura 2: Fichero de entrenamiento con los atributos U, V, PW. Cada línea de este fichero contiene el valor horario correspondiente a las predicciones meteorológicas U, V y producción de la energía o potencia eléctrica PW, es decir, cada línea contiene el valor correspondiente a los atributos U, V y PW para cada fecha y hora, desde el periodo "2009-07-01-00h" hasta "2010-12-31-12h".

Ejemplo de fichero en formato ".arff" con la segunda estructura:

@relation 'entrenUV1-weka.filters.unsupervised.attribute.Remove-R1-3,5-weka.filters.unsupervised.attribute.Remove-R1'

@attribute U numeric
@attribute V numeric

@attribute pw numeric

@data
2.34,-0.79,0.085
2.18,-0.99,0.02
2.2,-1.21,0.06
....

Estructura 3: Fichero de entrenamiento con los atributos WS, WD, PW. Cada línea de este fichero contiene el valor horario correspondiente a las predicciones meteorológicas WS, WD y producción de la energía o potencia eléctrica PW, es decir, cada línea contiene el valor correspondiente a los atributos WS, WD y PW para cada fecha y hora, desde el periodo "2009-07-01-00h" hasta "2010-12-31-12h".

Ejemplo de fichero en formato ".arff" con la tercera estructura:

@relation enterWSWD1-weka.filters.unsupervised.attribute.Remove-R1-5

@attribute Ws numeric
@attribute Wd numeric
@attribute pw numeric

@data
2.47,108.68,0.085
2.4,114.31,0.02
2.51,118.71,0.06
….

De los conjuntos de datos generados, se utiliza como conjunto de entrenamiento los datos que corresponden a los primeros 20 días de cada mes desde la fecha inicial 01/07/2009 hasta el 31/12/2010.

3.1.2. Conjunto de entrenamiento para la segunda parte.

Para la segunda parte de este trabajo se usan cuatro ficheros de entrenamiento para cada una de las granjas, en función del número de valores de instantes anteriores de la producción, con la siguiente estructura:

Fichero con datos que contiene los mejores atributos o estructura que resultan del estudio de la primera parte (U, V, WS, WD), el atributo "HORS" (la hora para la que se realiza la predicción), el atributo PW que es la producción de la energía en el instante t o "HORS" y hasta cuatro atributos con el valor de instantes anteriores de la producción de energía eléctrica (t1, t2, t3, t4), donde t1 es el valor de la producción de la energía (PW) en el instante anterior t-1; t2 es el valor de la producción de la energía (PW) en el instante anterior t-2; t3 es el de la producción de la energía (PW) en el instante anterior t-3; t4 es el valor de la producción de la energía (PW) en el instante anterior t-4.

Se ha implementado varios métodos mediante lenguaje de programación Java que nos permite generar un fichero genérico con un número determinado valores de instantes anteriores de la producción.

Ejemplo de fichero genérico con valores de cuatro instantes anteriores de la producción:

@relation entrenamiento
@attribute fecha numeric
@attribute mes numeric
@attribute dia numeric
@attribute hora numeric
@attribute horaPrediccion numeric
@attribute Ws numeric
@attribute Wd numeric
@attribute t1 numeric
@attribute t2 numeric
@attribute t3 numeric
@attribute t4 numeric
@attribute pw numeric
@data
2009,07,01,12,1,2.85,103.17,**0.085,0.02,0.06,0.045**,**0**
2009,07,01,12,2,3.2,103.36,0.085,0.02,0.06,0.045,**0.01**
2009,07,01,12,3,3.35,100.63,0.085,0.02,0.06,0.045,**0**
…

2009,07,01,12,45,2.74,99.14,0.085,0.02,0.06,0.045,0.025
2009,07,01,12,46,3.7,98.82,0.085,0.02,0.06,0.045,0.04
2009,07,01,12,47,4.84,98.94,0.085,0.02,0.06,0.045,0.1
2009,07,01,12,48,5.7,99.9,**0.085,0.02,0.06,0.045**,0.175
2009,07,02,00,1,1.27,147.93,**0,0.01,0,0**,0
2009,07,02,00,2,1.59,163.36,0,0.01,0,0,0.01
2009,07,02,00,3,1.46,175.48,0,0.01,0,0,0.01
…

Los cuatro ficheros de entrenamiento se generan a partir del fichero genérico, con la ayuda de la herramienta Weka, eliminando los atributos no necesarios para formar el conjunto de entrenamiento en cada caso. Por ejemplo, para generar el fichero con valores de dos instantes anteriores de la producción, se elimina del fichero genérico los siguientes atributos: fecha, mes, día, hora, t3, t4 y se guarda el fichero con su correspondiente identificativo.

Ejemplo de fichero con datos de entrenamiento con valores de dos instantes anteriores de la producción:

@relation 'entrenamiento-weka.filters.unsupervised.attribute.Remove-R1-4,10-11'

@attribute horaPrediccion numeric
@attribute Ws numeric

@attribute Wd numeric
@attribute t1 numeric
@attribute t2 numeric
@attribute pw numeric

@data
1,2.85,103.17,**0.085,0.02**,**0**
2,3.2,103.36,0.085,0.02,**0.01**
3,3.35,100.63,0.085,0.02,0
…

46,3.7,98.82,0.085,0.02,0.04
47,4.84,98.94,0.085,0.02,0.1
48,5.7,99.9,**0.085,0.02**,0.175
1,1.27,147.93,**0,0.01**,0
2,1.59,163.36,0,0.01,0.01
3,1.46,175.48,0,0.01,0.01
….

De los diferentes conjuntos de datos generados, se utiliza como conjunto de entrenamiento los datos que corresponden a los primeros 20 días de cada mes desde la fecha inicial 01/07/2009 hasta el 31/12/2010.

3.1.3. Conjunto de entrenamiento para la tercera parte.

Para la tercera parte de este trabajo se generas tres ficheros para cada una de las granjas, en función del horizonte temporal o "HORS". Como se ha comentado anteriormente en la descripción de los datos, para cada una de las granjas se registran predicciones cada doce horas para las cuarenta y ocho horas siguientes, por lo cual en la tercera parte del trabajo se decide estudiar diferentes modelos en función del número de horizonte temporal.

Para la generación de estos conjuntos se divide el conjunto de entrenamiento disponible para la segunda parte en tres conjuntos según el número de horizonte temporal u hora a la que se realiza la predicción. El primer conjunto contiene los datos que corresponde a 1 hasta 12 horas de predicción, el segundo conjunto de entrenamiento corresponde a los datos desde 12 hasta 24 horas de predicción y el tercer conjunto de entrenamiento corresponde a los datos desde el horizonte 24 hasta 48 horas.

A continuación se muestra un volcado de un fichero ejemplo de cada tipo:

Conjunto de entrenamiento para horizonte de 1 a 12: Datos de entrenamiento que contiene los mejores atributos o estructura que resultan del estudio de la segunda parte, que corresponden a cada uno de los doce horizontes temporales, es decir entradas de datos que corresponden a "HORS" igual a 1 hasta 12 horas, ejemplo:

@relation entrenamiento-weka.filters.unsupervised.attribute.Remove-R1-4

@attribute horaPrediccion numeric
@attribute Ws numeric

@attribute Wd numeric
@attribute t1 numeric
@attribute pw numeric

@data
1,2.85,103.17,0.085,0
2,3.2,103.36,0.085,0.01
3,3.35,100.63,0.085,0
…
9,1.67,114.58,0.085,0
10,1.44,124.63,0.085,0.005
11,1.22,135.16,0.085,0.015
12,1.04,155.24,0.085,0
1,1.27,147.93,0,0
2,1.59,163.36,0,0.01
3,1.46,175.48,0,0.01
…
9,0.88,172.7,0,0.015
10,1.08,185.05,0,0.03
11,1.28,199.95,0,0.01
12,1.44,203.82,0,0

1,2.22,169.75,0,0
2,2.4,162.17,0,0
3,2.52,158.7,0,0
…..

Conjunto de entrenamiento para horizonte de 12 a 24: Datos de entrenamiento que contiene los mejores atributos o estructura que resultan del estudio de la segunda parte, que corresponden a cada uno de los doce siguientes horizontes temporales, es decir entradas de datos que corresponden a "HORS" igual a 12 hasta 24 horas, ejemplo:

@relation entrenamiento-weka.filters.unsupervised.attribute.Remove-R1-4

@attribute horaPrediccion numeric
@attribute Ws numeric
@attribute Wd numeric
@attribute t1 numeric
@attribute pw numeric

@data
12,1.04,155.24,0.085,0
13,1.13,187.75,0.085,0
14,1.46,209.04,0.085,0.01
…
22,1.27,206.32,0.085,0.03
23,1.49,229.38,0.085,0.01

24,1.63,238.87,0.085,0
12,1.44,203.82,0,0
13,1.51,191.55,0,0
14,1.67,171.57,0,0
…
21,4.33,145.78,0,0.08
22,4.21,145.51,0,0.025
23,3.8,144.44,0,0.08
24,3.27,141.68,0,0.201
12,3.62,126.63,0,0.201
13,3.21,124.46,0,0.12
14,2.77,122.46,0,0.035
….

Conjunto de entrenamiento para horizonte de 24 a 48: Datos de entrenamiento que contiene los mejores atributos o estructura que resultan del estudio de la segunda parte, que corresponden a cada uno de veinte y cuatro últimos horizontes temporales, es decir entradas de datos que corresponden a "HORS" igual a 24 hasta 48 horas, ejemplo:

@relation entrenamiento-weka.filters.unsupervised.attribute.Remove-R1-4

@attribute horaPrediccion numeric
@attribute Ws numeric
@attribute Wd numeric
@attribute t1 numeric
@attribute pw numeric

@data
24,1.63,238.87,0.085,0
25,1.46,232.75,0.085,0
26,1.19,211.12,0.085,0
…
46,3.7,98.82,0.085,0.04
47,4.84,98.94,0.085,0.1
48,5.7,99.9,0.085,0.175
24,3.27,141.68,0,0.201
25,2.83,136.19,0,0.12
26,2.54,128.18,0,0.035
....

De los conjunto de datos generados, se utilizan como conjuntos de entrenamiento los datos que corresponden a los primeros 20 días de cada mes desde la fecha inicial 01/07/2009 hasta el 31/12/2010.

3.2. Generación del conjunto de test.

En este trabajo se realiza el test utilizando dos conjuntos diferentes de datos. **El primer conjunto de test** está formado por los datos correspondientes a los diez últimos días de cada mes, desde el 01/07/2009 hasta el 31/12/2010 y se utiliza en la fase de validación de los modelos, para determinar cuáles son los mejores modelos y los parámetros óptimos para los métodos de regresión, el Perceptrón Multicapa, el M5P.

Para la generación de este primer conjunto de test para cada una de las aproximaciones estudiadas en este trabajo, basta partir del conjunto de entrenamiento generado para cada una de las partes, que contiene la información diaria desde 01/07/2009 hasta el 31/12/2010, y extraer los datos relativos a los 10 últimos días de cada mes.

El **segundo conjunto de test** está compuesto por los datos desde el 01/01/2011 hasta el 26/06/2012. Estos datos contienen solo las entradas cuyas salidas están el fichero llamado "test.csv". Este conjunto de test fue el proporcionado por la competición y se utiliza para evaluar los mejores modelos generados y validados anteriormente con el primer conjunto de test y comparar las diferentes aproximaciones estudiadas en este trabajo.

Para utilizar los datos de test proporcionados por la competición (en este trabajo se le denota como "test real"), es necesario desarrollar métodos en Java para darles un formato entrada/salida para cada aproximación estudiada. Los conjuntos de test generados tendrán las entradas correspondientes para cada aproximación. A continuación se muestra un volcado de datos de los conjuntos de test real para cada una de las partes estudiadas.

Primera parte: Variables de entrada WS y WD

@relation test-weka.filters.unsupervised.attribute.Remove-R1-5

@attribute Ws numeric
@attribute Wd numeric
@attribute pw numeric

@data
5.05,37.88,0.471
4.89,38.38,0.491
4.82,37.83,0.551
…

Segunda Parte: Variables de entrada WS y WD y valores en instantes anteriores de la producción:

@relation TestReal-weka.filters.unsupervised.attribute.Remove-R5

@attribute horaPrediccion numeric
@attribute Ws numeric
@attribute Wd numeric

@attribute t1 numeric
@attribute pw numeric

@data
1,5.05,37.88,0.481,0.471
2,4.89,38.38,0.481,0.491
3,4.82,37.83,0.481,0.551
….

Tercera Parte: En ese caso es necesario generar tres ficheros de test real para cada uno de los tres modelos (modelo para predecir de 1 a 12 horas; modelo para predecir de 12 a 24 horas; modelo para predecir de 24 a 48 horas).

Primer conjunto de test real:

@relation TestReal-weka.filters.unsupervised.attribute.Remove-R5

@attribute horaPrediccion numeric
@attribute Ws numeric
@attribute Wd numeric
@attribute t1 numeric
@attribute pw numeric

@data
1,5.05,37.88,0.481,0.471
2,4.89,38.38,0.481,0.491
3,4.82,37.83,0.481,0.551
…
10,4.87,9.46,0.481,0.647
11,5.17,4.1,0.481,0.586
12,5.51,1.99,0.481,0.456
1,4.51,73.52,0.085,0.145
2,5.01,82.94,0.085,0.145
3,5.32,86.68,0.085,0.155
….

Segundo conjunto de test real:

@relation TestReal-weka.filters.unsupervised.attribute.Remove-R5

@attribute horaPrediccion numeric
@attribute Ws numeric
@attribute Wd numeric
@attribute t1 numeric
@attribute pw numeric

@data
12,5.51,1.99,0.481,0.456
13,5.86,3.64,0.481,0.581

14,6.29,7.06,0.481,0.541
…
23,6.13,64.72,0.481,0.762
24,6.08,61.99,0.481,0.707
12,3.77,136.33,0.085,0.105
13,4.21,137.31,0.085,0.095
14,4.65,135.16,0.085,0.14
15,4.82,130.45,0.085,0.13
….

Tercer conjunto de test real:

@relation TestReal-weka.filters.unsupervised.attribute.Remove-R5

@attribute horaPrediccion numeric
@attribute Ws numeric
@attribute Wd numeric
@attribute t1 numeric
@attribute pw numeric

@data
24,6.08,61.99,0.481,0.707
25,6.19,64.74,0.481,0.827
26,6.42,70.37,0.481,0.757
…
46,7.92,77.65,0.481,0.471
47,7.79,78.18,0.481,0.486
48,7.57,78.8,0.481,0.481
24,4.2,102.8,0.085,0.1
25,4.48,102.52,0.085,0.09
26,4.9,104,0.085,0.18
….

3.3. Lenguaje de programación utilizado

Para la generación de los diferentes conjuntos de datos (entrenamiento y test) se han desarrollado funciones en el lenguaje de programación JAVA. Se han desarrollado también métodos auxiliares para tratar los datos, calcular las medias de los cuarenta y ocho horizontes temporales y asignar a los resultados (salidas de los diferentes modelos) formato compatible con la herramienta Excel, para posteriormente poder analizar los resultados mediante gráficos de forma más eficiente y cómoda.

Actualmente existen varias alternativas para implementar las funciones necesarias para tratar los datos, entre estas alternativas se puede utilizar el lenguaje "M" y utilizar MATLAB y sus librerías que proporcionan diferentes métodos de aprendizaje automático.

También se puede utilizar el lenguaje R, que es un lenguaje y entorno de programación para análisis estadístico, gráfico y es útil para minería de datos.

Para realizar este trabajo se ha decidido utilizar el lenguaje JAVA, debido a que es un lenguaje de programación orientado a objetos, es un lenguaje intuitivo, proporciona una biblioteca de clases que facilitan desarrollar funciones para tratar ficheros de datos (API de JAVA) y es un lenguaje de programación que ha sido muy utilizado durante esta carrera y con el que estoy muy familiarizada.

Capítulo 4. Estudio de diferentes aproximaciones para predecir la producción de energía eólica

Durante este capítulo se van a validar diferentes alternativas de diseño de la solución al problema de la predicción de la energía eólica en un horizonte de predicción de 48 horas. La primera alternativa consiste en utilizar un modelo global que se construya a partir de variables meteorológicas, la segunda alternativa consiste en utilizar un modelo global que incorpore valores anteriores de la serie de producción y la última alternativa consiste en construir tres modelos para tres bloques diferentes de horizontes de predicción.

4.1 Parte I. Predicción de la producción de energía eólica a partir de predicciones meteorológicas.

El objetivo de esta primera parte del trabajo es validar diferentes modelos cambiando las variables de entrada al modelo.

Estos modelos siguen las siguientes ecuaciones:

- Modelo 1: Las variables de entrada para el primer modelo siguen la siguiente función: F1 (u, v, ws, wd).
- Modelo 2: Las variables de entrada para el segundo modelo siguen la siguiente función: F2 (u, v).
- Modelo 3: Las variables de entrada para el segundo modelo siguen la siguiente función: F3 (ws, wd).

Donde:

U: Es el componente del vector velocidad en el eje x.

V: Es el componente del vector velocidad en el eje y.

Ws: Es el módulo de la velocidad del viento.

Wd: Es la dirección del viento.

Cada una de las funciones F1, F2 y F3 comentadas anteriormente se va a aproximar con un modelo lineal, con un PM y con M5P, con el objetivo de estudiar el comportamiento de estos métodos de regresión y obtener el mejor modelo para predecir la producción de energía eólica.

4.1.1. Descripción de los experimentos realizados.

La primera parte de este trabajo consiste en determinar qué modelo de los expuestos anteriormente es el más adecuado para la resolución del problema de predicción de la energía eólica. Para ello se realizarán experimentos utilizando los diferentes métodos, el lineal, El Perceptrón Multicapa y el M5P, para validar cada uno de los tres modelos para cada una de las siete granjas eléctricas.

También se realizará un estudio de los parámetros para el método Perceptrón Multicapa y el M5P, para determinar cuáles son los parámetros son más adecuados para aproximar las funciones f1, f2, y f3.

Para el Perceptrón Multicapa, los parámetros que se variará su valor para el estudio de parámetros es el número de neuronas ocultas, se probara primeramente el valor de número de neuronas ocultas por defecto que es (número de variables de entrada (atributos) más el número de variables de salida o clases)/2, y se aumentará dicho valor, sumando al valor inicial el valor 5.

Los parámetros para el PM en función de los tres modelos:

Para el modelo 1 con F1, se probará con número de neuronas ocultas igual a 3, y 8.

Para el modelo 2 con F2 y el modelo 3 con F3 se probará con número de neuronas ocultas igual a 2 y 7.

Para el M5P, se realizará el estudio de parámetros variando el valor de dos parámetros, el BuildRegresion y Unpruned, que recibirán valores como verdadero o falso, consisten en la construcción del árbol de regresión o no, y podar el árbol o no.

En total se realizarán cuatro experimentos por cada modelo, de la siguiente forma:

Modelo 1, con F1, se probará con buildRegresion = true y unpruned=true, buildRegresion = true y unpruned=false, buildRegresion = false y unpruned=true y por ultimo con buildRegresion = false y unpruned=false.

Modelo 2, con F2, se probará con buildRegresion = true y unpruned=true, buildRegresion = true y unpruned=false, buildRegresion = false y unpruned=true y por ultimo con buildRegresion = false y unpruned=false.

Modelo 3, con F3, se probará con buildRegresion = true y unpruned=true, buildRegresion = true y unpruned=false, buildRegresion = false y unpruned=true y por ultimo con buildRegresion = false y unpruned=false.

Los modelos se entrenan con los datos correspondientes a los veinte primeros días de cada mes y se validan con los diez últimos días de cada mes. Y finalmente en función del error de validación (error medio absoluto obtenido en el test con los datos de los 10 últimos días de cada mes), elegiremos el mejor modelo de los tres modelos comentados anteriormente y la mejor configuración de parámetros para Perceptrón Multicapa y el M5P.

Y a continuación se usará este mejor modelo elegido anteriormente para realizar el test real, que consiste en el test con datos reales que se utilizó en la competición que ofrece la página web donde se descargó todos los datos utilizados en la experimentación.

Finalmente se va a hacer un estudio del mejor modelo por horizonte de predicción, con el fin de saber cómo se comportan los modelos para horizonte cercanos y lejanos, es decir, estudiar cómo evoluciona el error a medida que el horizonte de predicción aumenta. El error por horizontes consiste en calcular la media del error en valor absoluto para cada uno de los cuarenta y ocho horizontes, entendiendo por el error en valor absoluto a la diferencia en valor absoluto entre la salida real y la salida predicha del modelo para cada grupo de horizonte.

4.1.2. Resultados de los experimentos.

En apartado se va mostrar los resultados obtenidos en el estudio realizado en la primera parte de este trabajo.

Primero se muestran los resultados para la aproximación lineal de los tres modelos, a continuación se muestran los resultados del Perceptrón Multicapa y el M5P con sus correspondientes estudios de parámetros. Y por último se muestra mediante gráficas el error medio en valor absoluto del mejor modelo en la predicción por horizontes.

4.1.2.1. Regresión lineal simple.

A continuación se presenta una tabla con el valor del error medio de validación (test con los diez últimos días de cada mes) obtenido con cada uno de los tres modelos y para cada una de las siete granjas eléctricas cuando se realiza una regresión lineal.

Tabla 1. Regresión Lineal. Error Medio Absoluto para cada modelo

Granja eléctrica	Modelo 1	Modelo 2	Modelo 3
Granja1	0.133	0.184	0.133
Granja2	0.1611	0.2374	0.1611
Granja3	0.1493	0.2727	0.1493
Granja4	0.1414	0.238	0.1414
Granja5	0.1409	0.2299	0.1409
Granja6	0.1273	0.222	0.1273
Granja7	0.1378	0.2538	0.1378
Error medio total	0.1415	0.2340	0.1415

Como se puede observar en la tabla anterior el error medio de validación obtenido con el método lineal para el modelo 1 y modelo 3 es el mismo, también es más bajo que el error obtenido en el modelo 2.

4.1.2.2. Perceptron Multicapa.

A continuación se presenta una tabla con el valor del error medio absoluto obtenido en la validación (test con datos de los diez últimos días de cada mes) de cada uno de los modelos correspondientes a cada una de las siete granjas, en función del número de neuronas ocultas que se utilizan para entrenar la red.

Tabla 2. Perceptron Multicapa. Error Medio Absoluto para cada modelo y configuración.

	Modelo 1		**Modelo 2**		**Modelo 3**	
	3 ocultas	8 ocultas	2 ocultas	7 ocultas	2 ocultas	7 ocultas
Granja1	0.2071	0.1934	0.2731	0.2695	0.2116	0.1971
Granja2	0.1647	0.1651	0.2378	0.202	0.1773	0.173
Granja3	0.1479	0.133	0.3164	0.2165	0.1583	0.1566
Granja4	0.1434	0.1329	0.3561	0.3496	0.2347	0.153
Granja5	0.1446	0.1499	0.2473	0.2433	0.1773	0.147
Granja6	0.1174	0.1624	0.2682	0.2648	0.1209	0.1169
Granja7	0.1176	0.1109	0.3473	0.342	0.1773	0.1162
Error medio por parámetros	0.1490	0.15	0.2924	0.27	0.18	0.1514

Como se puede observar en la tabla anterior el error medio más bajo de validación obtenido con el método del Perceptron Multicapa es el error medio (0.1490) obtenido en el modelo 1 con tres neuronas ocultas.

4.1.2.3. M5P.

A continuación se presenta los resultados del error medio obtenido en la validación (test con datos de los diez últimos días de cada mes) aplicando el método M5P a los tres modelos en función de los parámetros buildRegresion y unpruned.

Tabla 3. M5P. Error Medio Absoluto para el modelo1 y diferentes configuraciones.

	Modelo 1			
	BR=true, UP=true	BR=true, UP=false	BR=false, UP=true	BR=false UP=false
Granja1	0.1324	0.1307	0.1334	0.1313
Granja2	0.1435	0.1429	0.1436	0.1413
Granja3	0.1352	0.1335	0.136	0.1338
Granja4	0.1306	0.1302	0.1313	0.1293
Granja5	0.126	0.1257	0.1265	0.1245
Granja6	0.1198	0.1183	0.122	0.1197
Granja7	0.1129	0.1119	0.1146	0.1109
Error medio por parámetros	0.1283	0.1277	0.1294	0.1274

Tabla 4. M5P. Error Medio Absoluto para modelo2 y diferentes configuraciones.

	Modelo 2			
	BR=true, UP=true	BR=true, UP=false	BR=false, UP=true	BR=false UP=false
Granja1	0.1341	0.1333	0.1338	0.1321
Granja2	0.145	0.1444	0.1445	0.1428
Granja3	0.141	0.1415	0.1392	0.1365
Granja4	0.1326	0.1329	0.1314	0.1296
Granja5	0.129	0.1296	0.1274	0.1264
Granja6	0.1213	0.1212	0.1219	0.1204
Granja7	0.1178	0.1182	0.1152	0.1141
Error medio por parámetros	0.1315	0.1316	0.1305	0.1288

Tabla 5. M5P. Error Medio Absoluto para modelo 3 y diferentes configuraciones.

	Modelo 3			
	BR=true, UP=true	BR=true, UP=false	BR=false, UP=true	BR=false UP=false
Granja1	0.132	0.1311	0.1327	0.1309
Granja2	0.1433	0.1429	0.1432	0.1417
Granja3	0.1351	0.1338	0.1362	0.1335
Granja4	0.1292	0.1291	0.1304	0.1288
Granja5	0.1261	0.1258	0.127	0.1253
Granja6	0.1193	0.1191	0.1204	0.1193
Granja7	0.1131	0.1127	0.1134	0.1115
Error medio por parámetros	0.1283	0.1278	0.129	0.1272

Observando los resultados obtenidos con el M5P en los tres modelos en función de los parámetros, se ve claramente que el modelo 1 y el modelo 3 obtienen casi los mismos resultados, el mejor parámetro es UP= false, BR= false con el que se ha obtenido un error medio del 0,1272.

4.1.3. Análisis de los resultados: el mejor modelo.

Analizando los resultados se observa que el Lineal y M5P obtienen los mismos resultados utilizando el modelo 1 con variables de entrada (U, V, WS, WD) y el modelo 3 con variables de entrada (WS, WD) y el Perceptrón Multicapa obtiene mejor resultado utilizando el modelo 1, pero el mejor resultado (0.1272) se ha obtenido aplicando el M5P al modelo 3 (WS, WD) con el parámetro UP= false y BR=false, por lo que se decide elegir como mejor modelo el modelo 3.

A continuación se presenta una tabla (tabla 6) con el resumen de los mejores resultados obtenidos para cada una de las granjas y para cada uno de los métodos de aproximación utilizados. En la tabla 7 se muestra un resumen de los mejores parámetros (número de neuronas ocultas) para el Perceptron Multicapa, y (podar o no podar y construir el árbol de regresión o no) para el M5P.

Tabla 6. Resumen del mejor modelo en la validación.

Granja	Lineal, test (últimos 10 días de cada mes).	Perceptron Multicapa, test (últimos 10 días de cada mes).	M5P, test (últimos 10 días de cada mes).
Granja1	0.133	0.1971	0.1309
Granja2	0.1611	0.173	0.1417
Granja3	0.1493	0.1566	0.1335
Granja4	0.1414	0.153	0.1288
Granja5	0.1409	0.147	0.1253
Granja6	0.1273	0.1169	0.1193
Granja7	0.1378	0.1162	0.1115
Error medio por parámetros	0.1415	0.1514	0.1272

Tabla 7. Resumen de los mejores parámetros para Perceptron Multicapa y el M5P.

Granja	Nº neuronas ocultas para el MLP.	BuildRegresion, Unpruned para el M5P.
Granja1	7	BR=true, UP=false.
Granja2	7	BR=false, UP=false.
Granja3	7	BR=true o false, UP=false.
Granja4	7	BR=true o false, UP=false.
Granja5	7	BR=false, UP=false.
Granja6	7	BR=true, UP=false.
Granja7	7	BR=false, UP=false.
Mejor parámetro	7	BR=false o true, UP=false.

Como se puede observar en la tabla 6 el mejor modelo resulta ser el tercer modelo formado por las dos variables de entrada WS, WD, que son variables que miden la temperatura y dirección del viento.

También se puede comprobar que el M5P es el método que mejor resultados ha obtenido para cada una de las granjas.

En cuanto a los mejores parámetros, para el Perceptrón Multicapa el mejor parámetro es un número de neuronas ocultas igual a siete, puesto que es el mejor parámetro utilizado en el modelo 3 (mejor modelo), mientras que el mejor parámetro para el

M5P es el BuildRegresion igual false y Unpruned igual a false (construir árbol de regresión o no, podar el árbol o no).

4.1.4. La evaluación de los modelos en el test de la competición.

La evaluación de los mejores modelos obtenidos mediante la Regresión Lineal Simple, Perceptron Multicapa y el M5P, se realiza utilizando el conjunto de test real (proporcionado por la competición), que está formado por el conjunto de datos reales desde el 01/11/2011 hasta el 26/06/2012. Esta evaluación se ha realizado con el modelo 3, que ha sido el que mejor resultados proporciona en validación. Para el PM se ha utilizado seis neuronas ocultas y para M5P se ha utilizado BuildRegresion igual a false y Unpruned igual a false, que es la mejor configuración de parámetros obtenida en la fase de validación.

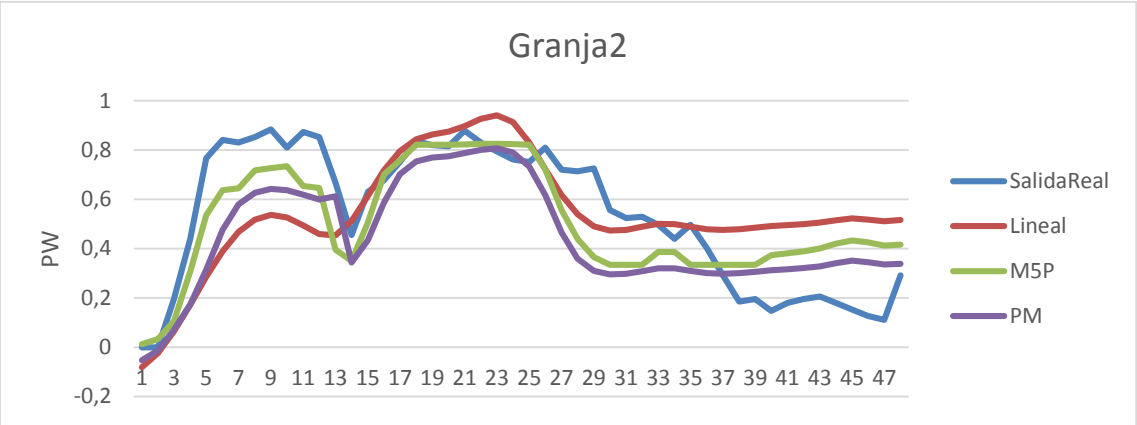
Tabla 8. Resumen del mejor modelo con test real.

Granja Eléctrica	Método Lineal	Método PM	Método M5P
Granja1	0.1404	0.1506	0.1394
Granja2	0.1589	0.1414	0.1332
Granja3	0.1627	0.1556	0.1477
Granja4	0.1515	0.1503	0.134
Granja5	0.1529	0.1461	0.1315
Granja6	0.1353	0.173	0.1236
Granja7	0.1411	0.1282	0.1138
Error medio total	0.1490	0.1493	0.1319

Observando los resultados obtenidos con los tres métodos se puede ver que existe una diferencia entre el resultado obtenido en test real el test de validación realizado anteriormente, los resultados obtenidos en la evaluación del mejor modelo con el test real no han sido mejores que los resultados obtenidos en la validación y que el método M5P sigue siendo el que mejor resultados obtiene.

Para observar la predicción de los diferentes métodos, en el gráfico 1 se muestra la salida real y la salida predicha para la granja 2 con los tres métodos de regresión utilizados y para el primer periodo de 48 horas.

Gráfico 1. Parte I. Salida real y predicha de los diferentes métodos.



Se observa que los tres métodos siguen la tendencia de la producción real, aunque no consiguen predecir con exactitud ciertos valores de la producción, también se observa que la predicción utilizando el M5P es más exacta que la predicción utilizando los demás métodos.

4.1.5. Estudio para cada uno de los 48 horizontes con el test real.

En esta sección se evaluará el error medio para cada horizonte temporal utilizando el mejor modelo (modelo 3), con el objetivo de analizar el comportamiento del método Lineal, el PM y el M5P y comprobar cómo evoluciona la predicción de la salida real, desde el primer horizonte temporal hasta el horizonte cuarenta y ocho.

A continuación se muestra mediante gráficos (Gráfico 1 hasta el Gráfico 7) el error medio cometido en el test real durante la predicción de la salida del modelo para cada horizonte desde 1 a 48 horas, para cada una de las granjas.

Gráfico 2. Parte I. Granja1. Error medio de diferentes métodos para cada uno de los 48 horizontes.

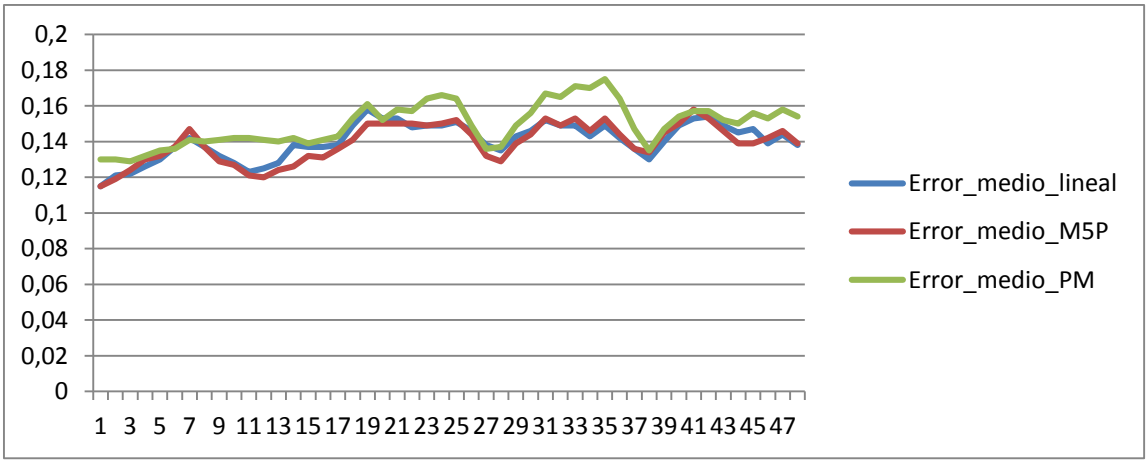


Gráfico 3. Parte I. Granja2. Error medio de diferentes métodos para cada uno de los 48 horizontes.

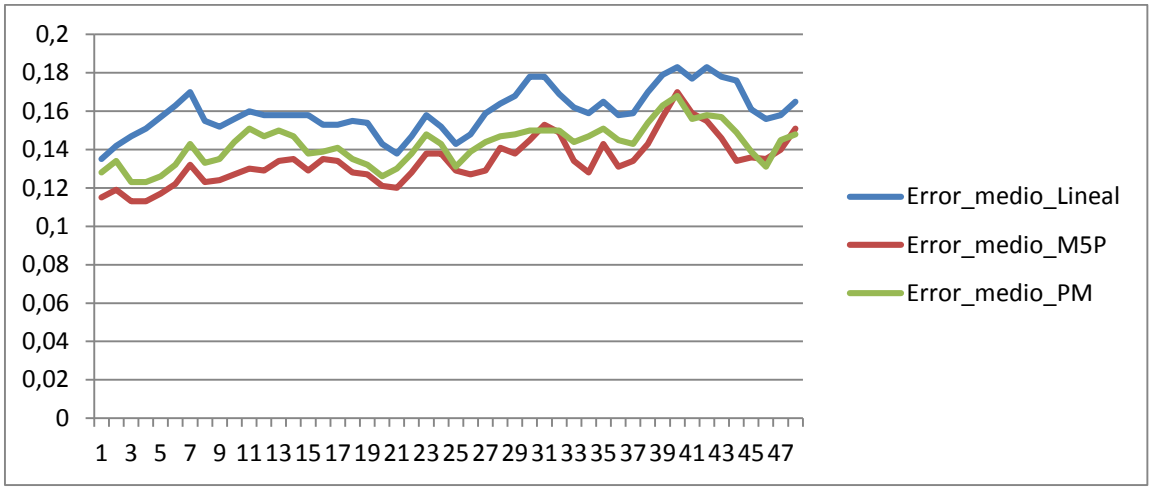


Gráfico 4. Parte I. Granja3. Error medio de diferentes métodos para cada uno de los 48 horizontes.

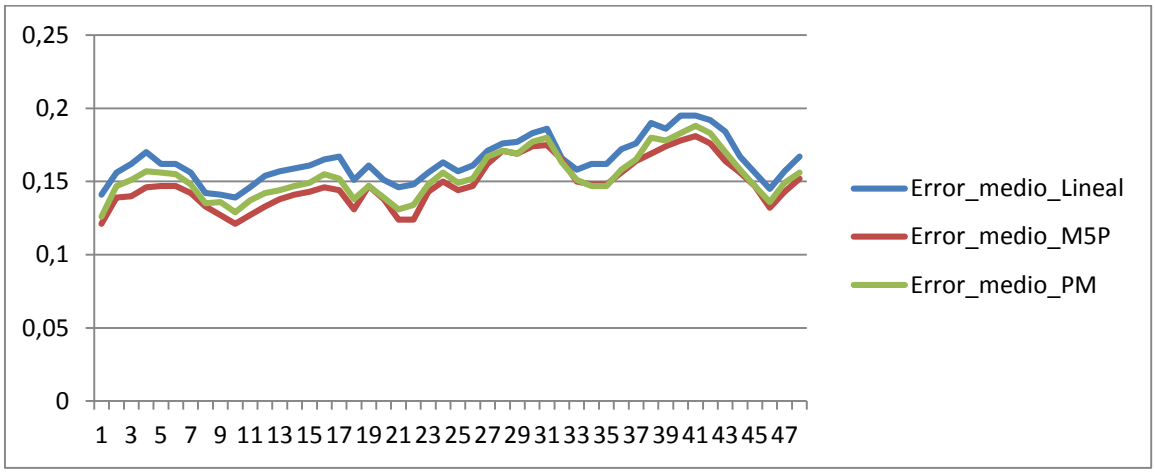


Gráfico 5. Parte I. Granja4. Error medio de diferentes métodos para cada uno de los 48 horizontes.

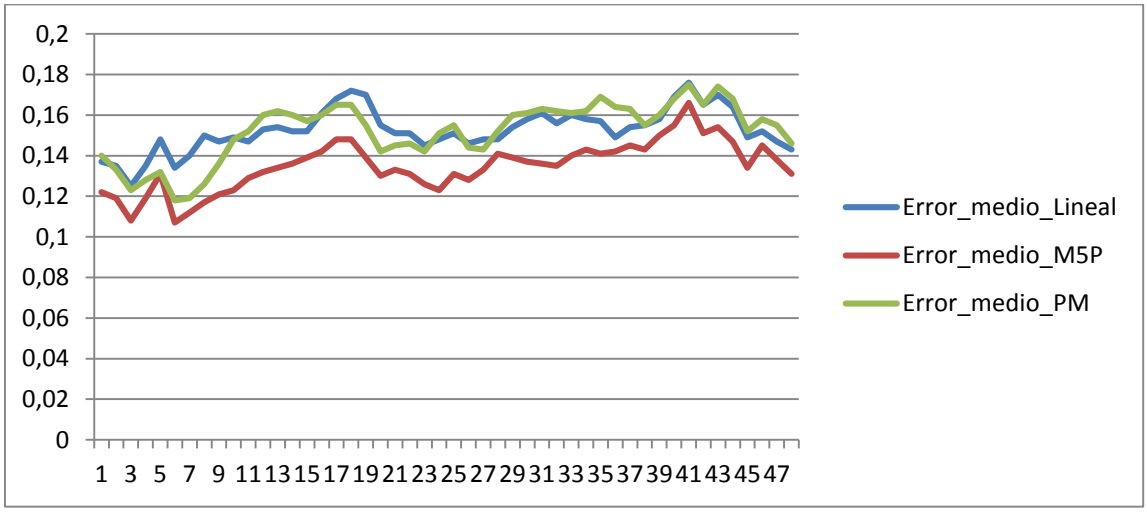


Gráfico 6. Parte I. Granja5. Error medio de diferentes métodos para cada uno de los 48 horizontes.

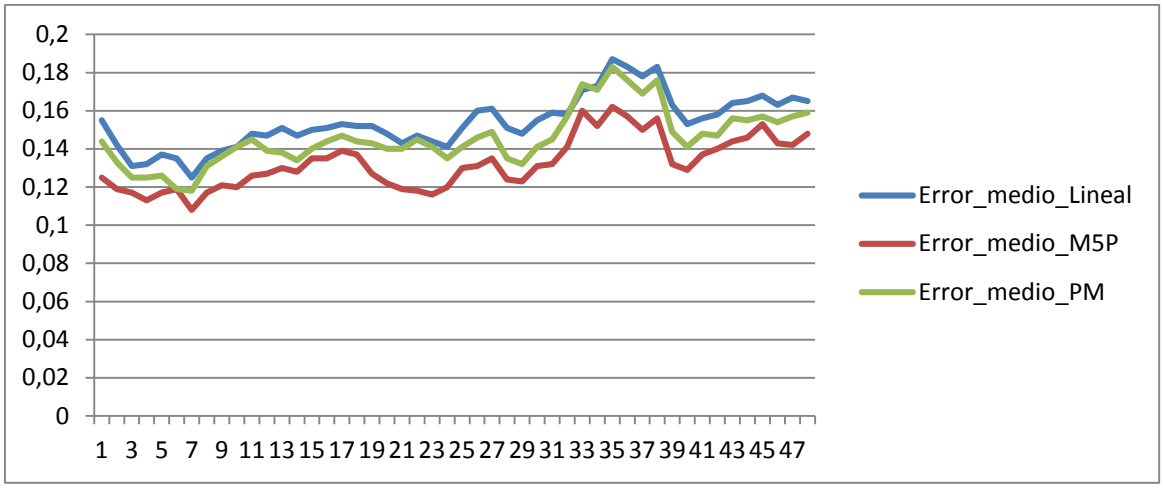


Gráfico 7. Parte I. Granja 6. Error medio de diferentes métodos para cada uno de los 48 horizontes.

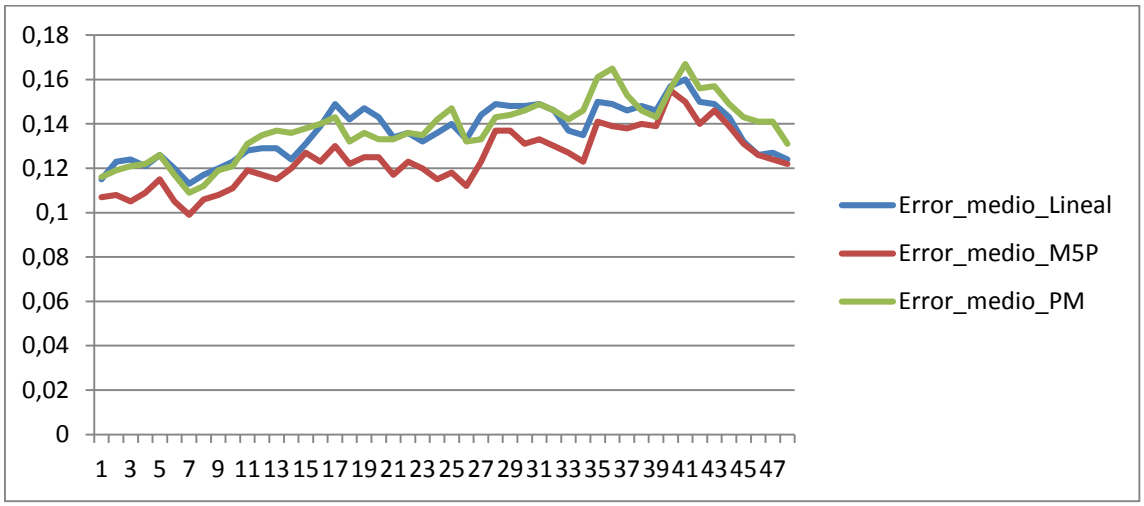
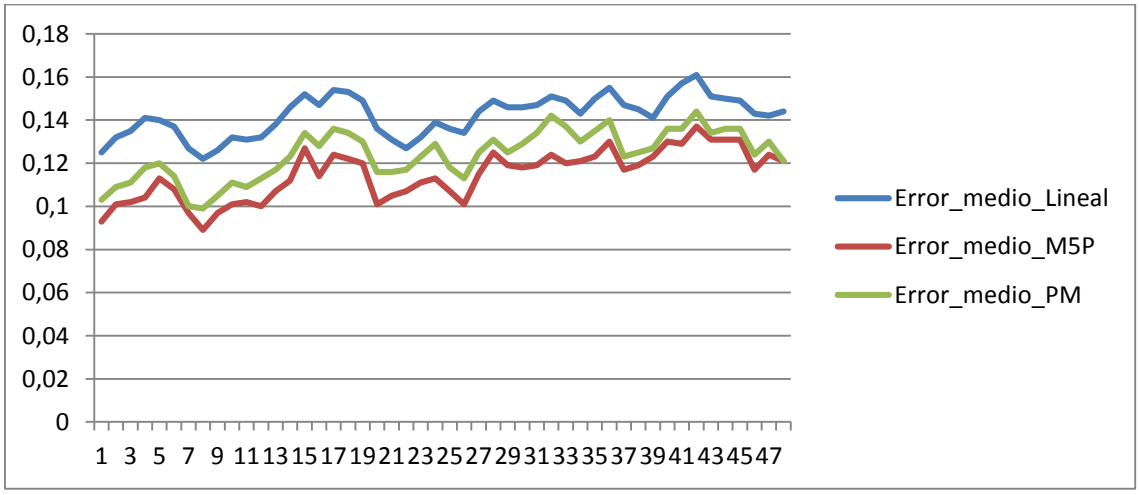


Gráfico 8. Parte I. Granja 7. Error medio de diferentes métodos para cada uno de los 48 horizontes.



Analizando los siete gráficos anteriores correspondientes a cada granja, se puede observar lo siguiente:

En el gráfico 1 correspondiente a la granja 1 se puede ver claramente que el modelo lineal es mejor que el Perceptron Multicapa, mientras que el M5P es el mejor, también se puede observar que el error para los primeros horizontes es más bajo que para los últimos.

En los gráfico 2, 3, 5 y 7 se observa claramente que el Perceptron multicapa resulta mejor que el lineal en los cuarenta y ocho horizontes, mientras que el que mejor resulta de los tres métodos es el M5P, también se puede ver error crece a medida que aumenta el horizonte temporal.

Y por último observando los gráficos 4 y 6 se observa que para algunos horizontes resulta mejor el Perceptron multicapa que el lineal, y para otros horizontes el lineal resulta mejor que el Perceptron multicapa, pero como mejor método resulta siempre el M5P. También se observa que siempre el error es más bajo para los primeros horizontes temporales y más alto para los últimos.

4.1.6. Conclusiones de los resultados.

En esta primera parte del trabajo se ha llegado a la conclusión de que el mejor modelo es el modelo 3 con variables de entrada WS, WD, para predecir la producción de la energía o potencia para el futuro. En general el mejor método de aproximación es el M5P y el error medio de la predicción de la salida real para los cuarenta y ocho horizontes es más bajo en los primeros que en los últimos horizontes de predicción.

4.2. Parte II. Predicción de la producción de energía eólica a partir de predicciones meteorológicas y de la producción

En esta segunda parte del trabajo construimos los modelos usando los métodos de regresión utilizados anteriormente, pero además incluimos en el modelo el horizonte temporal y valores anteriores disponibles de la serie temporal para luego predecir un horizonte de 1 a 48 horas. El objetivo es estudiar cómo influye el uso de valores o instantes anteriores de la producción en la predicción.

Se parte del modelo 3 que era el que mejor resultados proporcionaba en el estudio realizado anteriormente (Parte I) con la siguiente ecuación:

WP(t+h)=F4(h,ws(t+h-1), wd(t+h-1), WP(t),…,WP(t-r))

Donde WP(t),…,WP(t-r) son los valores de la producción en los instantes de tiempo t, t-1, …,t-r; h es el horizonte temporal (1,…,48); ws(t+h-1) es la predicción de la velocidad del viento para el instante t, t+1,...; y wd(t+h-1) es la predicción de la dirección del viento para el instante t, t+1,...

Además de estudiar y analizar el comportamiento de los diferentes métodos (lineal, PM y M5P), también se pretende estudiar el número de instantes anteriores de la serie.

Una vez construido el modelo utilizando los diferentes métodos de regresión, dicho modelo será utilizado para predecir el horizonte de 1 a 48, del siguiente modo:

WP(t+1)=F4(1,ws(t), wd(t), WP(t),…,WP(t-r))

WP(t+2)=F4(2, ws(t+1), wd(t+1), WP(t),…,WP(t-r))

…

WP(t+48)=F4(48, ws(t+47), wd(t+47), WP(t),…,WP(t-r))

Los valores ws(t+1),… ws(t+48), wd(t+1),… wd(t+48) vienen dados en los datos de la competición.

4.2.1. Descripción de los experimentos realizados

Realizaremos un estudio de los mejores parámetros para el Perceptron Multicapa (número de neuronas ocultas igual a 1, 7, 10, 15, 20), y los mejores parámetros para el M5P (BuildRegresion y Unpruned), que recibirán valores como verdadero o falso, consisten en construcción del árbol de regresión o no y podar el árbol de regresión o no.

Para el estudio de parámetros se utilizará el modelo WP(t+h) para r igual a tres, es decir, este modelo tendrá como variables de entrada el horizonte (HORS), la velocidad del viento(WS), la dirección del viento (WD), primer instante anterior (WP (t-1)), segundo instante anterior (WP(t-2)) y el tercer instante anterior (WP(t-3)). El modelo se entrena con los datos correspondientes a los veinte primeros días de cada mes y se valida con los diez últimos días de cada mes. Y finalmente en función del error de validación (error medio absoluto obtenido en el test con los datos de los 10 últimos

días de cada mes), elegiremos la mejor configuración de parámetros para Perceptron Multicapa y el M5P. El estudio de parámetros se realizará para cada una de las granjas.

Una vez realizado el estudio de los mejores parámetros, realizaremos pruebas para diferentes valores de r o instantes anteriores (r=1, 2, 4). De esta forma aplicaremos el método lineal, el Perceptron Multicapa y el M5P para los tres modelos (modelo 1 con r= 1; modelo 2 con r=2; modelo 3 con r=4). Estos tres modelos se entrenan con los datos correspondientes a los veinte primeros días de cada mes y se validan con los diez últimos días de cada mes, y para estos tres modelos no realizaremos el estudio de parámetros, usaremos los mejores parámetros que hemos obtenido en el estudio realizado anteriormente con el modelo con tres instantes anteriores como variables de entrada al modelo. El estudio de modelos con diferentes números de instantes anteriores como variables de entrada al modelo se realizará para cada una de las siete granjas.

A continuación compararemos entre los resultados obtenidos en los cuatro modelos estudiados anteriormente (modelo con r=1, modelo con r=2, modelo con r=3, modelo con r=4) y seleccionaremos el mejor modelo. Como medida de precisión nos centraremos en el error medio absoluto cometido a la hora de predecir la salida del modelo.

Después se utilizará el mejor modelo para predecir el horizonte de 1 a 48, para cada una de las granjas, para ello se realizará el test real (test que nos proporciona la competición) para evaluar este mejor modelo y determinar cuál es el método que cuya salida del modelo se aproxima mejor a la salida real.

Finalmente se realizará un estudio del mejor modelo por horizonte de predicción para cada una de las siete granjas, con el fin de saber cómo se comportan los modelos para horizontes cercanos y lejanos, es decir, estudiar cómo evoluciona el error a medida que el horizonte de predicción aumenta. Consiste en calcular la media del error en valor absoluto para cada uno de los cuarenta y ocho horizontes temporales, entendiendo por el error en valor absoluto a la diferencia en valor absoluto entre la salida real y la salida predicha del modelo para cada horizonte.

4.2.2. Resultados de los experimentos

En apartado se va mostrar los resultados obtenidos en el estudio realizado en la segunda parte de este trabajo.

Primero se muestra los resultados del Perceptron Multicapa y el M5P con sus correspondientes estudio de parámetros. Y por último se muestra mediante gráficas el error medio en valor absoluto del mejor modelo en la predicción por horizontes.

4.2.2.1. Perceptron Multicapa.

A continuación se presenta una tabla con el valor del error medio absoluto obtenido en validación (test con los últimos 10 días de cada mes) para cada una de las granjas eléctricas, en función del número de neuronas ocultas que se utilizan para entrenar la red.

En este estudio se ha utilizado únicamente un modelo con tres instantes anteriores y se ha realizado diferentes pruebas aplicando el Perceptrón Multicapa con 2, 7, 10, 15 y 20 neuronas ocultas.

Tabla 9. Perceptrón Multicapa. Error Medio Absoluto para cada configuración. Tres instantes anteriores.

Granjas eléctricas	2 oculta	7 ocultas	10 ocultas	15 ocultas	20 ocultas
Granja1	0.1298	0.1278	0.1279	0.1279	0.1296
Granja2	0.1556	0.1547	0.1537	0.1527	0.1581
Granja3	0.1761	0.1568	0.1579	0.1576	0.1551
Granja4	0.1478	0.1306	0.1304	0.13	0.1296
Granja5	0.142	0.1427	0.1416	0.1416	0.1416
Granja6	0.122	0.1148	0.1165	0.116	0.1175
Granja7	0.1287	0.1157	0.1205	0.1211	0.1219
Error total medio	0.1431	0.1347	0.1355	0.1353	0.1362

Como se puede observar en la tabla 9, el método del Perceptron Multicapa consigue mejor resultado con un número de neuronas ocultas igual a siete, ya que para la mayoría de las granjas se consigue el mejor resultado con un número de neuronas igual a siete, también la media del error de aproximación del modelo para las siete granjas resulta del 0.1347, es el valor del error más bajo conseguido en comparación con los demás casos de estudios.

4.2.2.2. M5P.

A continuación se presenta una tabla con los resultados del error medio obtenido en la validación (test con los últimos diez días de cada mes) para cada una las granjas, aplicando el método M5P en función de los dos parámetros buildRegresion y unpruned.

Para realizar este estudio se ha utilizado un único modelo con tres instantes anteriores (modelo WP(t+h), para r=3), y cada uno de los dos parámetros recibe un valor igual a verdadero o falso.

Tabla 10. M5P. Error Medio Absoluto para cada configuración. Tres instantes anteriores.

Granjas eléctricas	BR=true, UP=true	BR=true, UP=false	BR=false, UP=true	BR=false UP=false
Granja1	0.1318	0.1311	0.1348	0.131
Granja2	0.1489	0.1478	0.1512	0.146
Granja3	0.1353	0.1335	0.1398	0.1364
Granja4	0.1294	0.128	0.1327	0.1285
Granja5	0.1295	0.1289	0.1331	0.1287
Granja6	0.1176	0.1161	0.1215	0.1182
Granja7	0.1137	0.1134	0.1161	0.1128
Error medio	0.1295	0.1284	0.1327	0.1288

Como se puede observar en la tabla 10, el método del M5P consigue los mejores resultados con el BuildRegresion igual a true y Unpruned igual a false, ya que para la mayoría de las granjas se consigue el mejor resultado con estos parámetros, también la media del error de aproximación del modelo para las siete granjas resulta del 0.1284, es el valor del error más bajo conseguido en comparación con los demás casos de estudios.

4.2.2.3. Los mejores parámetros para el Perceptrón Multicapa y el M5P.

A continuación se presenta una tabla (tabla11) donde se resumen los mejores resultados obtenidos para cada una de las granjas y para cada una de las configuraciones utilizadas de los métodos de aproximación utilizados.

Tabla 11. Resumen de los mejores resultados con tres instantes anteriores.

Granja	Modelo Lineal, test (10 últimos días de cada mes).	Modelo P. Multicapa, test (10 últimos días de cada mes).	Modelo M5P, test (10 últimos días de cada mes).
Granja1	0.1335	0.1278	0.1311
Granja2	0.1608	0.1547	0.1478
Granja3	0.1493	0.1568	0.1335
Granja4	0.1413	0.1306	0.128
Granja5	0.1408	0.1427	0.1289
Granja6	0.1271	0.1148	0.1161
Granja7	0.1375	0.1157	0.1134
Error medio total.	0.1414	0.1347	0.1284

Observando el error medio en el test del conjunto con tres instantes anteriores, (tabla 10), se puede observar que el modelo del M5P consigue los mejores resultados (menor error medio) con el parámetro buildRegresion igual a true y unpruned igual a false y el modelo del Perceptrón Multicapa consigue mejor resultado utilizando siete neuronas ocultas (tabla 9).

En la tabla 12 se muestra un resumen de los mejores parámetros o configuración para el Perceptrón Multicapa (número de neuronas ocultas) y para el M5P (construir el árbol de regresión (buildRegresion), podar el árbol (unpruned)), para cada una de las granjas.

Tabla 12. Resumen de los mejores parámetros para el M5P, PM.

Granja	El modelo M5P.	El modelo Perceptron Multicapa, nº neuronas ocultas.
Granja1	BR=false, UP=false.	7
Granja2	BR=false, UP =false.	15
Granja3	BR =true, UP =false.	20
Granja4	BR=true, UP =false.	20
Granja5	BR=false, UP =false.	10,20,15
Granja6	BR=true, UP =false.	7
Granja7	BR=false, UP=false.	7
Mejor parámetro	BR=true, UP =false.	7

Como se puede observar el mejor parámetro para el Perceptrón Multicapa para la mayoría de las granjas es de siete neuronas ocultas y para el M5P la mejor configuración es buildRegresion igual a true y Unpruned igual a false, por lo que se decide elegir dichos parámetros para realizar los experimentos restantes (con una, dos y cuatro instantes anteriores). Los mejores parámetros se han sido seleccionados en función del error medio total obtenido en las granjas.

4.2.2.4. Estudio utilizando diferente número de instantes anteriores como entrada al modelo.

A continuación se muestran los resultados de regresión lineal utilizando un diferente número de valores anteriores, también se incluyen los resultados del mejor modelo de la parte 1, es decir cuando el número de instantes anteriores es cero. El error que se muestra en la tabla 13 es el error de validación (últimos 10 días de cada mes) para cada una de las granjas.

Tabla 13. Regresión Lineal. Error Medio Absoluto utilizando diferente número de valores anteriores.

Granja eléctrica	0 instantes anteriores	1 instante anterior	2 instantes anteriores	3 instantes anteriores	4 instantes anteriores
Granja1	0.133	0.1335	0.1335	0.1335	0.1335
Granja2	0.1611	0.1608	0.1608	0.1608	0.1608
Granja3	0.1493	0.1493	0.1493	0.1493	0.1493
Granja4	0.1414	0.1413	0.1413	0.1413	0.1413
Granja5	0.1409	0.1408	0.1408	0.1408	0.1408
Granja6	0.1273	0.1271	0.1271	0.1271	0.1271
Granja7	0.1378	0.1375	0.1375	0.1375	0.1375
Error medio total	0.1415	0.1414	0.1414	0.1414	0.1414

Como se puede observar en la tabla anterior el uso de valores de instantes anteriores para aproximar el modelo con el método Lineal aporta una pequeña mejora a los resultados conseguidos en la parte 1, ya que se consigue el mismo resultado para cada una de las granjas (0,1414) y es un resultado mejor que en el caso de usar el modelo sin variables de entrada con valores de instantes anteriores.

A continuación se muestra los resultados del Perceptrón Multicapa utilizando diferentes números de valores anteriores. Se muestra el error de validación (últimos 10 días de cada mes), para cada una de las granjas.

Tabla 14. Perceptrón Multicapa. Error Medio Absoluto utilizando diferente número de valores anteriores.

Granja eléctrica	0 instantes anteriores	1 instante anterior	2 instantes anteriores	3 instantes anteriores	4 instantes anteriores
Granja1	0.1971	0.1276	0.1279	0.1278	0.1287
Granja2	0.173	0.1554	0.156	0.1547	0.1535
Granja3	0.1566	0.149	0.1522	0.1568	0.1594
Granja4	0.153	0.1295	0.132	0.1306	0.1311
Granja5	0.147	0.1349	0.136	0.1427	0.1459
Granja6	0.1169	0.1176	0.1179	0.1148	0.1146
Granja7	0.1162	0.1197	0.1152	0.1157	0.1174
Error medio total	0.1514	0.1334	0.1339	0.1347	0.1358

Observando la tabla 14 se puede comprobar que el mejor modelo para el Perceptrón Multicapa es el modelo con variables de entrada con valores de un único instante anterior (error medio del 0.1334), también se puede observar que incluir valores de instantes anteriores como variables de entrada al modelo mejora bastante los resultados.

Tabla 15. M5P. Error Medio Absoluto utilizando diferente número de valores anteriores.

Granja eléctrica	0 instantes anteriores	1 instante anterior	2 instantes anteriores	3 instantes anteriores	4 instantes anteriores
Granja1	0.1309	0.1303	0.1318	0.1311	0.1321
Granja2	0.1417	0.1443	0.1437	0.1478	0.1451
Granja3	0.1335	0.1336	0.1341	0.1335	0.1342
Granja4	0.1288	0.1283	0.1284	0.128	0.1283
Granja5	0.1253	0.1267	0.1279	0.1289	0.1296
Granja6	0.1193	0.1184	0.1188	0.1161	0.1177
Granja7	0.1115	0.113	0.1127	0.1134	0.1138
Error medio total	0.1272	0.1278	0.1282	0.1284	0.1287

Observando la tabla 15 se puede comprobar que el mejor modelo para el M5P es el modelo con variables de entrada con valores de un único instante anterior (error medio del 0.1278), pero también se puede observar que incluir valores de instantes anteriores como variables de entrada al modelo no aporta mejoras a los resultados, puesto que con 0 instantes se obtiene un error menor (0.1272).

A continuación se presenta una tabla que resume el error medio total en la validación (diez últimos días de cada mes) para cada una de las granjas, en función del número de instantes anteriores que forman parte de las variables de entradas del modelo. También incluimos los resultados del mejor modelo de la parte 1, es decir cuando el número de instantes anteriores es cero.

Tabla 16. Resumen del error medio total en función del número de valores de instantes anteriores.

Nº de valores de instantes anteriores.	Lineal, test (10 últimos días de cada mes).	P. Multicapa, test (10 últimos días de cada mes).	M5P, test (10 últimos días de cada mes).
0	0.1415	0.1514	0.1272
1	0.1414	0.1334	0.1278
2	0.1414	0.1339	0.1282
3	0.1414	0.1347	0.1284
4	0.1414	0.1358	0.1287

Para el método Lineal el mejor modelo es el modelo con uno o varios instantes anteriores, mientras para el Perceptrón multicapa es con un instante anterior. Para el M5P no es positivo añadir instantes anteriores, pero en el resto del capítulo se analizarán los resultados de M5P con un instante anterior, puesto que la diferencia con 0 instantes es pequeña y el estudio para 0 instantes ya se hizo en la primera parte.

Tabla 17. Resumen del mejor modelo con 1 instante anterior.

Granja	Modelo Lineal, test (10 últimos días de cada mes).	Modelo P. Multicapa, test (10 últimos días de cada mes).	Modelo M5P, test (10 últimos días de cada mes).
Granja1	0.1335	0.1276	0.1303
Granja2	0.1608	0.1554	0.1443
Granja3	0.1493	0.149	0.1336
Granja4	0.1413	0.1295	0.1283
Granja5	0.1408	0.1349	0.1267
Granja6	0.1271	0.1176	0.1184
Granja7	0.1375	0.1197	0.113
Error medio total.	0.1414	0.1334	0.1278

Observando la tabla con el resumen del mejor modelo con instantes anteriores como variables de entradas al modelo, se puede comprobar que el mejor método de

regresión es el M5P, con el modelo con un único instante anterior como variables de entrada ha conseguido un error medio del 0.1278.

4.2.3. Evaluación de los mejores modelos con Test Real.

La evaluación de los mejores modelos obtenidos mediante la Regresión Lineal Simple, Perceptron Multicapa y el M5P, se realiza utilizando el conjunto de test real (proporcionado por la competición), que está formado por el conjunto de datos reales desde el 01/11/2011 hasta el 26/06/2012. Esta evaluación se ha realizado con el modelo con valores de un único instante anterior como variables de entrada, que ha sido el que mejor resultados proporciona en validación. Para el PM se ha utilizado siete neuronas ocultas y para M5P, se ha utilizado BuildRegresion igual true y Unpruned igual a false, que es la mejor configuración de parámetros obtenida en la validación.

Tabla 18. Resumen del mejor modelo con test real y un instante anterior.

Granja Eléctrica	Método Lineal	Método PM	Método M5P, BR=true, UP= false
Granja1	0.1404	0.1641	0.139
Granja2	0.1589	0.2162	0.133
Granja3	0.1627	0.1629	0.149
Granja4	0.1515	0.1394	0.1357
Granja5	0.1529	0.1425	0.133
Granja6	0.1353	0.1202	0.1272
Granja7	0.1411	0.1575	0.1149
Error medio total	0.1490	0.1575	0.1331

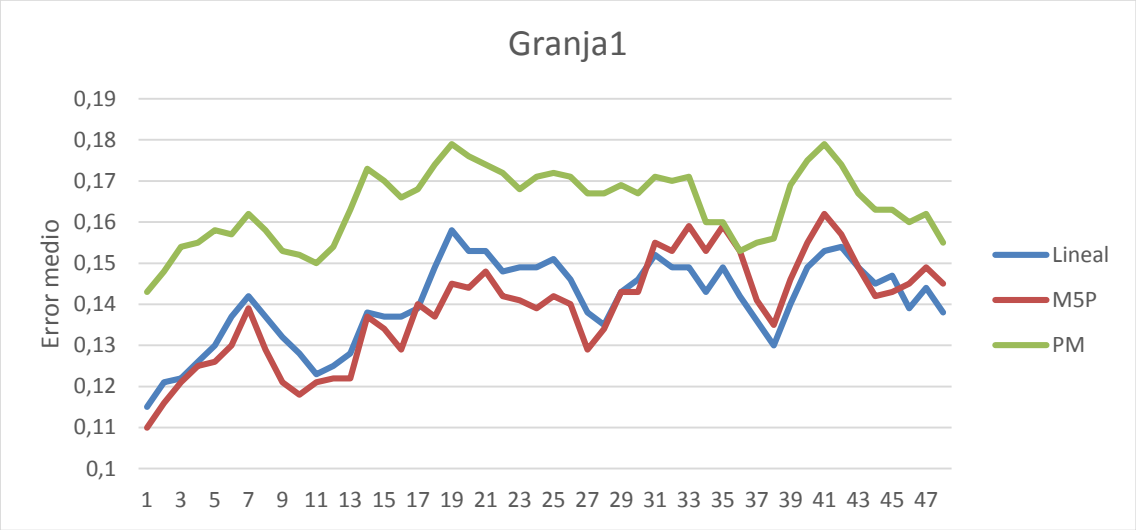
Observando los resultados obtenidos con los tres métodos se puede ver que el error medio de los dos métodos, regresión Lineal y el Perceptrón Multicapa ha aumentado en la evaluación del modelo con el test real y el método M5P ha obtenido mejor resultado en la evaluación con el test real que en validación y por último sigue siendo el que mejor resultados obtiene.

4.2.4. Estudio para cada uno de los 48 horizontes con Test Real.

En esta sección se evaluará el error medio para cada horizonte temporal utilizando el mejor modelo (modelo con un único instante anterior como valores de variables de entrada), con el objetivo de analizar el comportamiento del método Lineal, el PM y el M5P y comprobar cómo evoluciona la predicción de la salida real desde el primer horizonte temporal hasta el horizonte cuarenta y ocho.

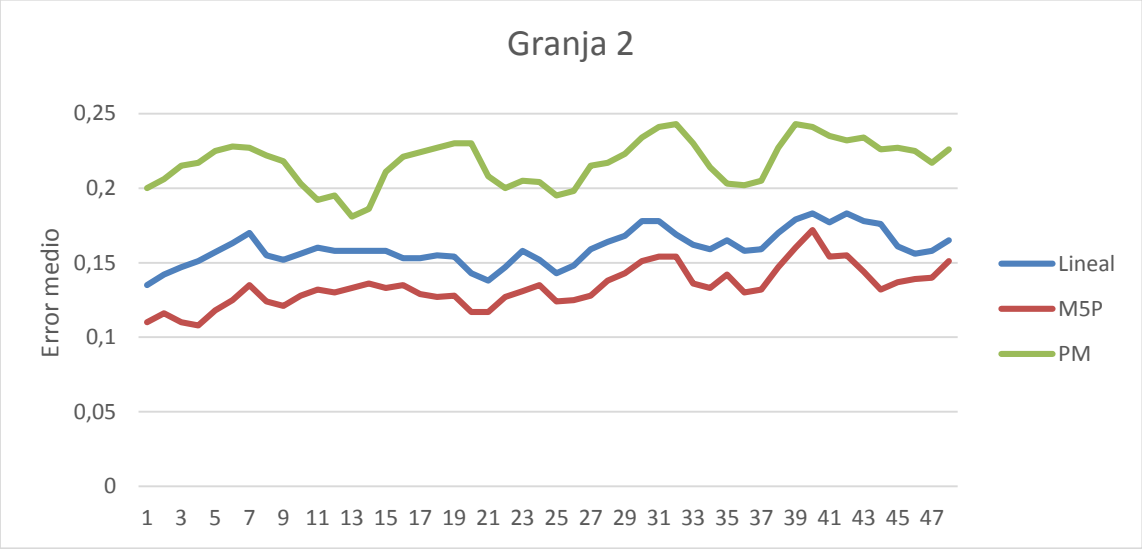
A continuación se muestra mediante gráficos (Gráfico 1 hasta la Gráfico 7) el error medio cometido en el test real durante la predicción de la salida del modelo para cada horizonte desde 1 a 48 horas, para cada una de las granjas.

Gráfico 9. Parte II. Granja 1. Error medio de diferentes métodos para cada uno de los 48 horizontes.



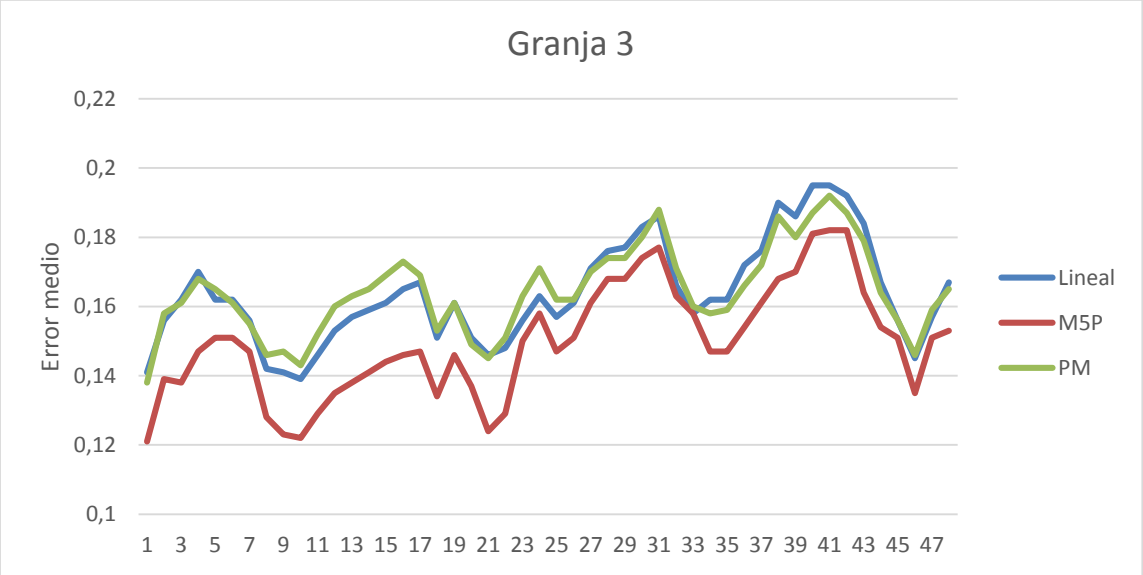
En el gráfico 9 correspondiente a la granja 1 se puede ver claramente para los primeros horizontes resulta mejor la aproximación de la salida del modelo con el método M5P, mientras que para los últimos horizontes resulta mejor el Lineal, la peor aproximación de la salida del modelo es la del Perceptron Multicapa, en general se puede observar que error medio de predicción para los primeros horizontes es más bajo que para los últimos.

Gráfico 10. Parte II. Granja 2. Error medio de diferentes métodos para cada uno de los 48 horizontes.



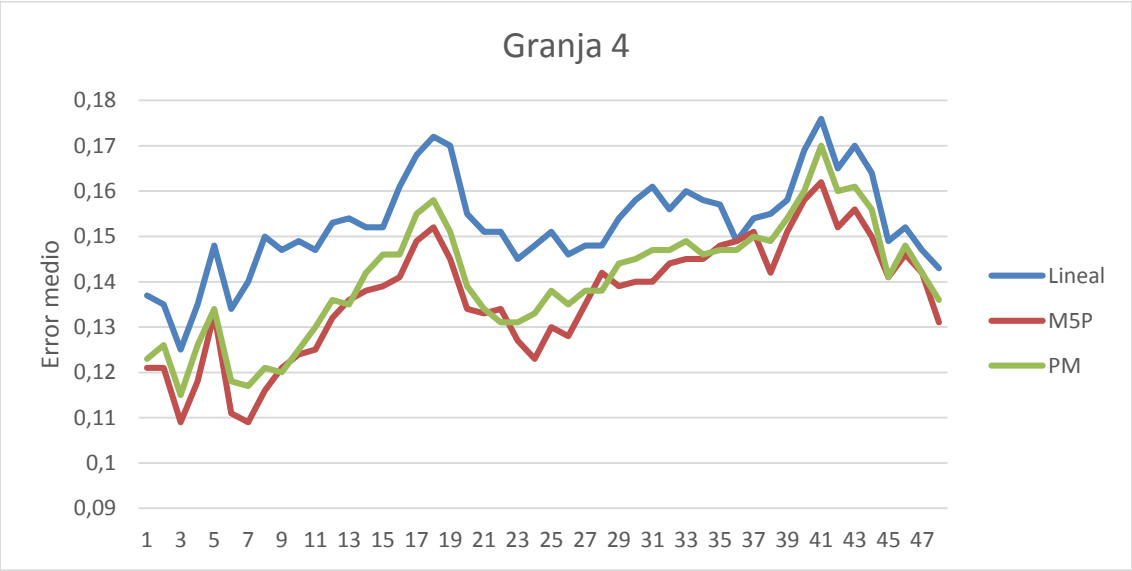
En el gráfico 10 correspondiente a la granja 2 se puede observar que el mejor método es el M5P y el peor es el Perceptron Multicapa en todos los horizontes, en general se puede observar que el error medio de predicción para los primeros horizontes es más bajo que para los últimos.

Gráfico 11. Parte II. Granja 3. Error medio de diferentes métodos para cada uno de los 48 horizontes.



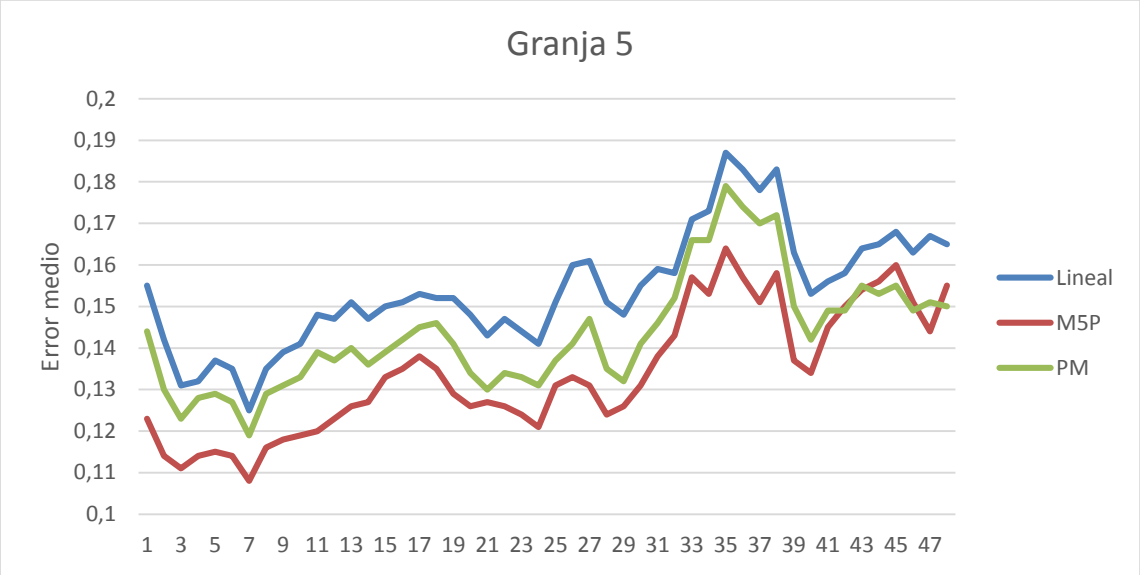
En el gráfico 11 correspondiente a la granja 3 se puede observar que el mejor método es el M5P, el Lineal y el Perceptron Multicapa obtienen resultados parecidos, en general se puede observar que el error medio de predicción para los primeros horizontes es más bajo que para los últimos.

Gráfico 12. Parte II. Granja 4. Error medio de diferentes métodos para cada uno de los 48 horizontes.



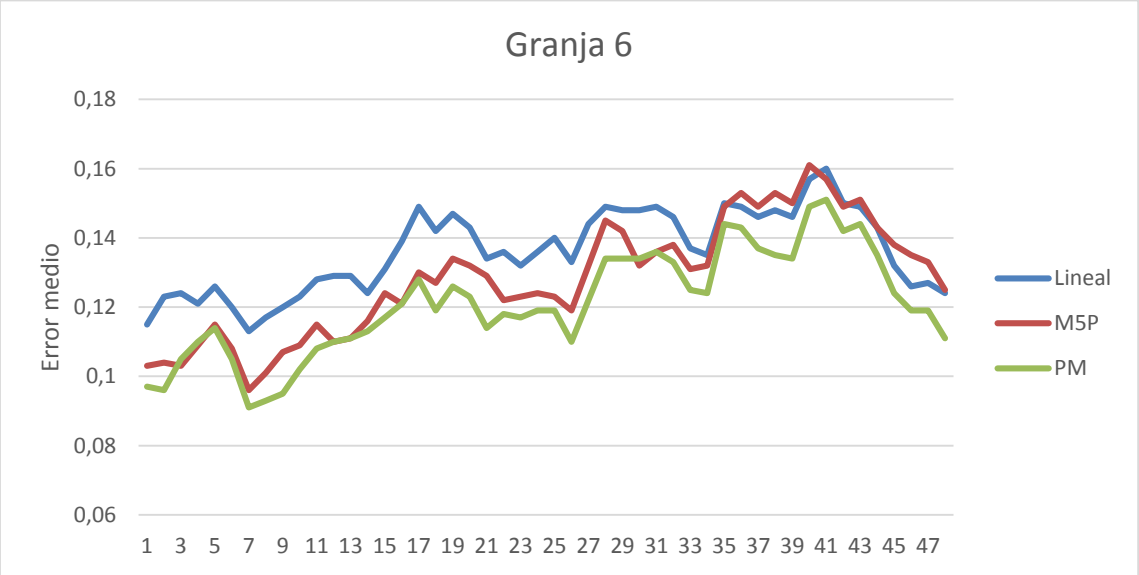
En el gráfico 12 correspondiente a la granja 4 se puede observar que los dos métodos el M5P y Perceptron Multicapa obtienen resultados muy parecidos, el Lineal obtiene peor resultados en todos los horizontes, en general se observa que el error medio de predicción para los primeros horizontes es más bajo que para los últimos.

Gráfico 13. Parte II. Granja 5. Error medio de diferentes métodos para cada uno de los 48 horizontes.



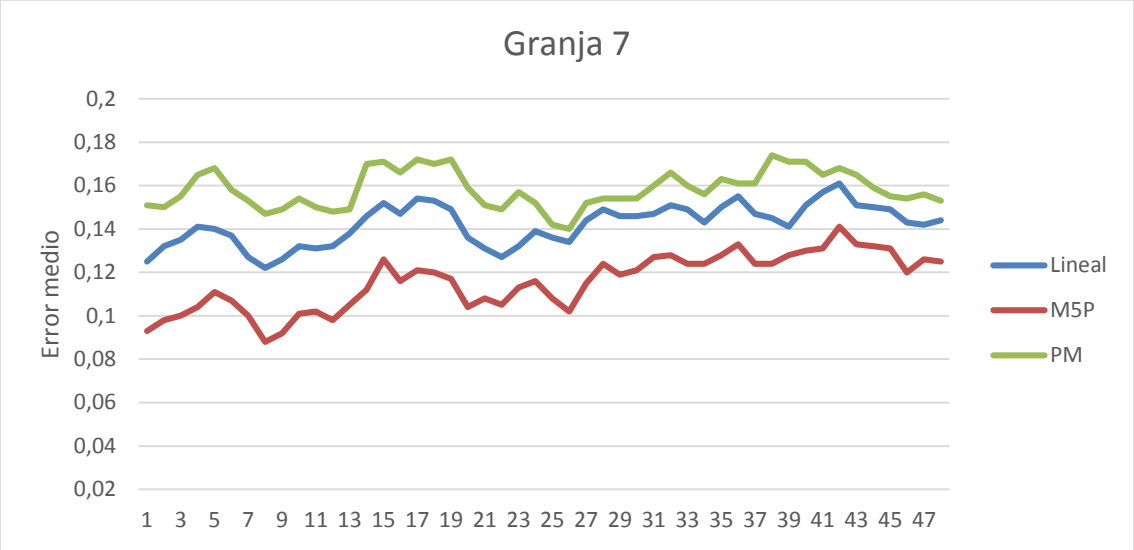
En el gráfico 13 correspondiente a la granja 5 se puede observar que el mejor método es el M5P, y el peor es el Lineal, en general se puede observar que el error medio de predicción para los primeros horizontes es más bajo que los últimos.

Gráfico 14. Parte II. Granja 6. Error medio de diferentes métodos para cada uno de los 48 horizontes.



En el gráfico 14 correspondiente a la granja 6 se puede observar que el mejor método es el Perceptron Multicapa, en general se puede observar que el error medio de predicción para los primeros horizontes es más bajo que los últimos y que los tres métodos consiguen resultados muy parecidos.

Gráfico 15. Parte II. Granja 7. Error medio de diferentes métodos para cada uno de los 48 horizontes.



En el gráfico 15 correspondiente a la granja 7 se puede observar que el mejor método es el M5P, y el peor es el Perceptron Multicapa, en general se puede observar que el error medio de predicción para los primeros horizontes es más bajo que los últimos.

4.2.5. Conclusiones de los resultados

En esta segunda parte del trabajo se ha llegado a la conclusión de que el uso de valores de instantes anteriores de la producción como variables de entrada al modelo para predecir la producción de la potencia para el futuro, permite la obtención de mejores resultados tanto en el modelo lineal como en el Perceptrón, si se usa un instante anterior. M5P no se beneficia de usar instantes anteriores, aunque la diferencia entre usar cero y un instante es muy pequeña. En cualquier caso, M5P sigue siendo el mejor modelo. Se ha comprobado que el error medio de la predicción de la salida real para los cuarenta y ocho horizontes es más bajo en los primeros que en los últimos horizontes.

4.3. Parte III. Predicción de la producción de energía eólica utilizando modelos con diferentes horizontes temporales

Esta última parte consiste en construir varios modelos para grupos de horizontes temporales de predicción. Concretamente, para cada uno de los tres métodos, el lineal, el PM y el M5P, construimos tres modelos para grupos de horizontes diferentes: el primer grupo es desde el horizonte 1 hasta el 12, el segundo grupo sería desde 13 hasta 24, y el último desde los 25 hasta 48.

Para construir los tres modelos con diferentes horizontes se parte del modelo 1 (modelo con un único valor de instantes anteriores de producción) que era el que proporcionaba resultados similares o mejores en el estudio realizado anteriormente (Parte II) con las siguientes ecuaciones:

Modelo 1: WP(t+h)=F4(h,ws(t+h-1), wd(t+h-1), WP(t)) , h pertenece al intervalo [1,12].

Modelo 2: WP(t+h)=F4(h,ws(t+h-1), wd(t+h-1), WP(t)), h pertenece al intervalo [13,24].

Modelo 3: WP(t+h)=F4(h,ws(t+h-1), wd(t+h-1), WP(t)), h pertenece al intervalo [25,48].

Donde WP(t) es la producción en el instante tiempo t; h es el horizonte temporal (en el modelo 1 equivale a (1,2,3,…,12), en el modelo 2 equivale a (13,14,…,24) y en el modelo 3 equivale a (25,26,…,48)); ws(t+h-1) es la velocidad del viento en el instante t+h+1; wd(t+h-1) es la dirección del viento en el instante t+h+1.

Se pretende estudiar y analizar el comportamiento de los diferentes métodos (lineal, PM y M5P), con los modelos con diferentes horizontes temporales y comparar los resultados obtenidos con modelos con el mismo horizonte temporal estudiados en la segunda parte de este trabajo.

Una vez construido los tres modelos con diferentes horizontes utilizando los diferentes métodos de regresión, dichos modelos serán utilizados para predecir el horizonte de 1 a 48, del siguiente modo:

El modelo 1 será utilizado para predecir el horizonte de 1 a 12 de la siguiente forma:

WP(t+1)=F4(1,ws(t), wd(t), WP(t))

WP(t+2)=F4(2, ws(t+1), wd(t+1), WP(t))

…

WP(t+12)=F4(12, ws(t+11), wd(t+11), WP(t))

El modelo 2 será utilizado para predecir el horizonte de 13 a 24 de la siguiente forma:

WP(t+13)=F4(13, ws(t+12), wd(t+12), WP(t))

WP(t+14)=F4(14, ws(t+13), wd(t+13), WP(t))

…

WP(t+24)=F4(24, ws(t+23), wd(t+23), WP(t))

El modelo 3 será utilizado para predecir el horizonte de 25 a 48 de la siguiente forma:

WP(t+25)=F4(25, ws(t+24), wd(t+24), WP(t))

WP(t+26)=F4(26, ws(t+25), wd(t+25), WP(t))

…

WP(t+48)=F4(48, ws(t+47), wd(t+47), WP(t))

Los valores ws(t+1),…, ws(t+48), wd(t+1),…, wd(t+48) vienen dados por la competición.

4.3.1. Descripción de los experimentos realizados

Se realizará la evaluación de los tres modelos (modelos con diferentes horizontes) mediante el test real, utilizando los tres métodos de regresión (Lineal, PM, M5P), para cada una de las granjas eléctricas.

En esta última parte no se realizan los estudios de los parámetros para el Perceptron Multicapa y el M5P, se utilizarán los mejores parámetros obtenidos en la validación en la segunda parte del trabajo.

Se utilizarán los tres métodos de regresión, Lineal, el Perceptron Multicapa y el M5P para evaluar los tres modelos utilizando el test Real (test de la competición), para el Perceptrón Multicapa se utilizarán siete neuronas ocultas y el M5P con BuildRegresion igual a true y Unpruned igual a false, que era la mejor configuración obtenida en la validación realizada en la segunda parte de este trabajo.

Se utilizarán los tres modelos para predecir el horizonte de 1 a 48 por separado, es decir el modelo 1 se utilizará para predecir el horizonte 1 a 12, el modelo 2 se utilizará para predecir el horizonte 13 a 24 y el modelo 3 se utilizará para predecir el horizonte 25 a 48, para cada una de las granjas, y se decide cuál es el método que cuya salida del modelo se aproxima mejor a la salida real.

Para decidir cuál es el mejor método y modelo se calcula la media del error utilizando los resultados obtenidos en los tres modelos estudiados anteriormente (modelo1 con horizonte desde 1 hasta 12 horas, modelo 2 con horizonte desde 13 hasta 24 horas y modelo 3 con horizonte desde 25 hasta 48 horas) y seleccionaremos el mejor modelo. Como medida de precisión nos centraremos en el error medio absoluto cometido a la hora de predecir la salida del modelo. Para calcular la media se utiliza la siguiente función de media ponderada:

(Error del Modelo1a12 en Test * NumDatosTestReal1a12 + Error del Modelo13a24 en Test * NumDatosTestReal13a24 + Error del Modelo25a48 en Test * Número de Datos Test Real 25 a 48) / Total de datos del Test real.

Se realizará un estudio del mejor modelo por horizonte de predicción para cada una de las siete granjas, con el fin de saber cómo se comportan los modelos para horizontes cercanos y lejanos, es decir, estudiar cómo evoluciona el error a medida que el horizonte de predicción aumenta. Consiste en calcular la media del error en valor absoluto para cada uno de los cuarenta y ocho horizontes temporales, entendiendo

por el error en valor absoluto a la diferencia en valor absoluto entre la salida real y la salida predicha del modelo para cada horizonte.

Compararemos entre los resultados obtenidos en los tres modelos con diferentes horizontes, los resultados obtenidos (modelos con el mismo horizonte temporal) en la segunda parte de este trabajo y los resultados obtenidos en primera parte de este trabajo, y decidiremos que modelos y métodos son más adecuados para resolver nuestro problema de predicción de la energía eólica.

4.3.2. Resultados de los experimentos

En apartado se van a mostrar los resultados obtenidos en el estudio realizado en la tercera parte de este trabajo.

Primero se muestran los resultados obtenidos en la evaluación de los tres modelos con diferentes horizontes utilizando los tres métodos de regresión, el Lineal, el Perceptrón Multicapa y el M5P con el test real.

Y finalmente se muestran los resultados obtenidos en el estudio de la media del error por horizontes de 48 horas.

4.3.2.1. Lineal Simple.

A continuación se presenta una tabla con el valor del error medio absoluto obtenido en la evaluación de los tres modelos para grupos de horizontes diferentes (test real) para cada una de las granjas eléctricas.

Tabla 19. Resumen de la evaluación de los tres modelos con el lineal.

Granjas eléctricas	Lineal Modelo para 1 a 12 Test real	Lineal Modelo para 13 a 24 Test real	Lineal Modelo para 25 a 48 Test real	Media de los 3 lineales Test real
Granja 1	0.1281	0.1431	0.145	0.1405
Granja 2	0.1515	0.1511	0.1646	0.1579
Granja 3	0.1507	0.1566	0.1722	0.163
Granja 4	0.1408	0.155	0.1549	0.1515
Granja 5	0.1396	0.1477	0.1631	0.1535
Granja 6	0.1209	0.1349	0.142	0.1351
Granja 7	0.131	0.1411	0.1471	0.1417
Error medio total	0.13751	0.14855	0.15936	0.14902

Observando los resultados obtenidos aplicando el método Lineal sobre los tres modelos se puede ver que el modelo para predecir 1 a 12 horizontes es mejor, (obtiene un error medio total de 0.13751) y que a medida que aumenta el horizonte de predicción es mayor el error de predicción.

4.3.2.2. Perceptrón Multicapa.

A continuación se muestran los resultados obtenido con el Perceptrón Multicapa con siete neuronas ocultas, para cada una de las granjas eléctricas.

Tabla 20. Resumen de la evaluación de los tres modelos con el Perceptrón Multicapa.

Granjas eléctricas	PM para 1 a 12 Test real	PM para 13 a 24 Test real	PM para 25 a 48 Test real	Media de los 3 PM Test real
Granja 1	0.1361	0.1403	0.1578	0.1480
Granja 2	0.1379	0.1361	0.1734	0.1552
Granja 3	0.1455	0.1394	0.1837	0.1630
Granja 4	0.1564	0.1447	0.1545	0.1524
Granja 5	0.1226	0.1789	0.1715	0.1617
Granja 6	0.1055	0.1408	0.1848	0.1543
Granja 7	0.1672	0.1163	0.1541	0.1474
Error medio total	0.13322	0.15095	0.173625	0.15457

Observando los resultados obtenidos aplicando el Perceptrón Multicapa sobre los tres modelos se puede ver que el modelo para predecir 1 a 12 horizontes es mejor, (obtiene un error medio total de 0.13322) y que a medida que aumenta el horizonte de predicción es mayor el error de predicción.

4.3.2.3. M5P.

Los resultados obtenidos con M5P (buildRegresion igual a true y unpruned igual a false) se muestran a continuación.

Tabla 21. Resumen de la evaluación de los tres modelos con el M5P.

Granjas eléctricas	M5P para 1 a 12. Test real	M5P para 13 a 24. Test real	M5P para 25 a 48. Test real	Media de los 3 M5P. Test real
Granja 1	0.1226	0.1389	0.1456	0.1383
Granja 2	0.1308	0.1303	0.1435	0.1370
Granja 3	0.1387	0.144	0.1636	0.1525
Granja 4	0.1194	0.1352	0.1457	0.1367
Granja 5	0.1204	0.1323	0.1468	0.1367
Granja 6	0.1099	0.1242	0.1378	0.1276
Granja 7	0.101	0.1173	0.128	0.1187
Error medio total	0.11788	0.13392	0.14458	0.13535

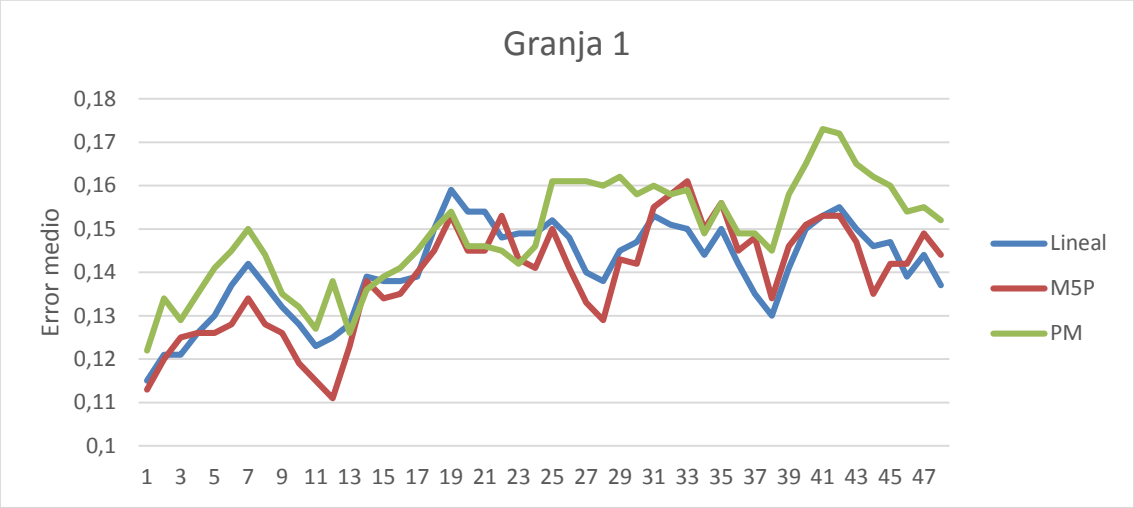
Observando los resultados obtenidos aplicando el M5P sobre los tres modelos se puede ver que el modelo para predecir 1 a 12 horizontes es mejor, (obtiene un error medio total de 0.11788) y que a medida que aumenta el horizonte de predicción es mayor el error de predicción.

4.3.3. Estudio para cada uno de los 48 horizontes con Test Real.

En esta sección se evaluará el error medio para cada horizonte temporal utilizando los tres modelos con el objetivo de analizar el comportamiento del método Lineal, el PLM y el M5P y comprobar cómo evoluciona la predicción de la salida real desde el primer horizonte temporal hasta el horizonte cuarenta y ocho.

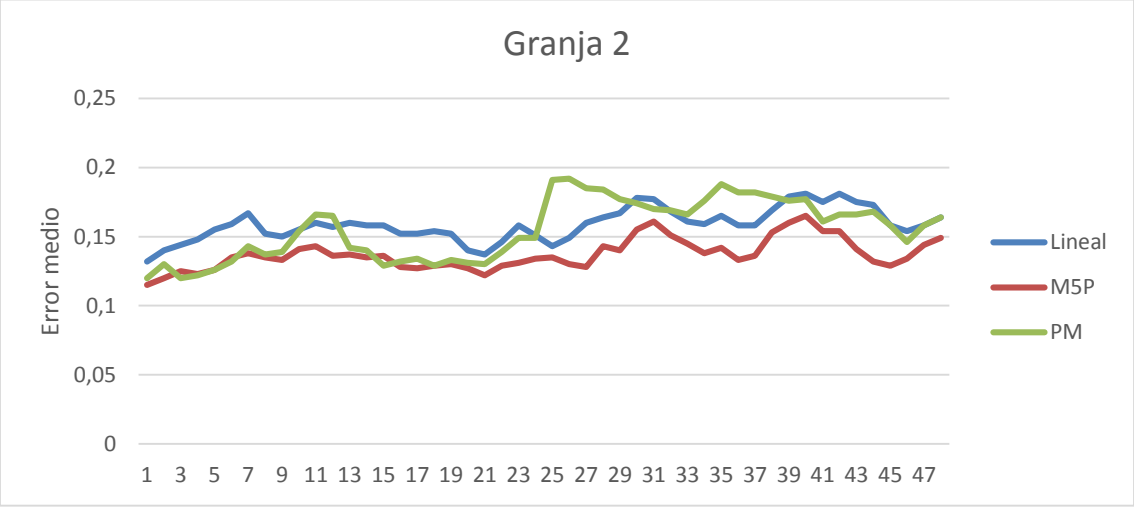
A continuación se muestra mediante gráficos (Gráfico 1 hasta el Gráfico 7) el error medio cometido en el test real durante la predicción de la salida del modelo para cada horizonte desde 1 a 48 horas, para cada una de las granjas.

Gráfico 16. Parte III. Granja 1. Error medio de diferentes métodos para cada uno de los 48 horizontes.



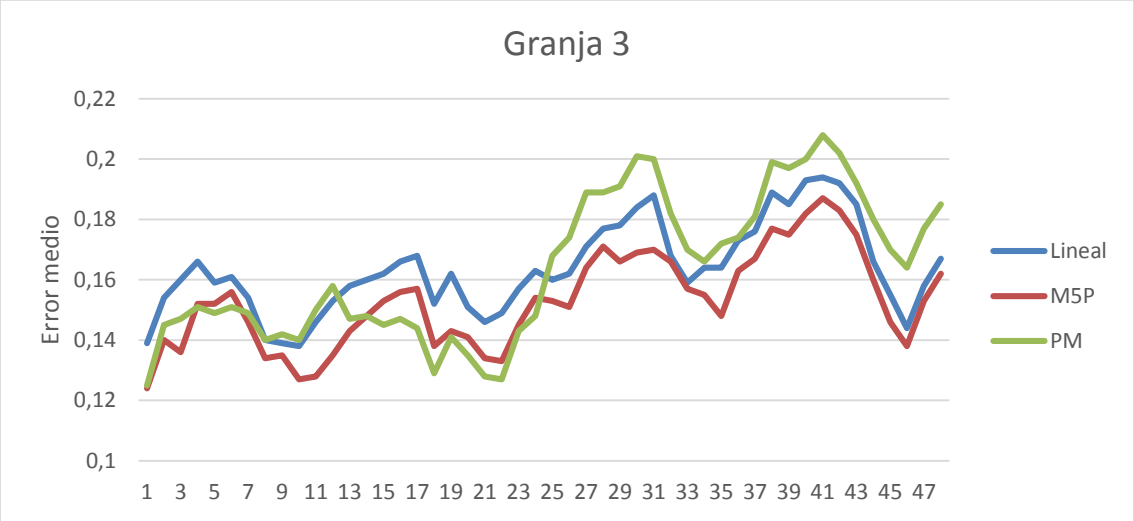
En el gráfico 16 correspondiente a la granja 1 se puede observar que el mejor método es el M5P, y el peor es el Perceptron Multicapa, en general se puede observar que el error medio de predicción para los primeros horizontes es más bajo que los últimos.

Gráfico 17. Parte III. Granja 2. Error medio de diferentes métodos para cada uno de los 48 horizontes.



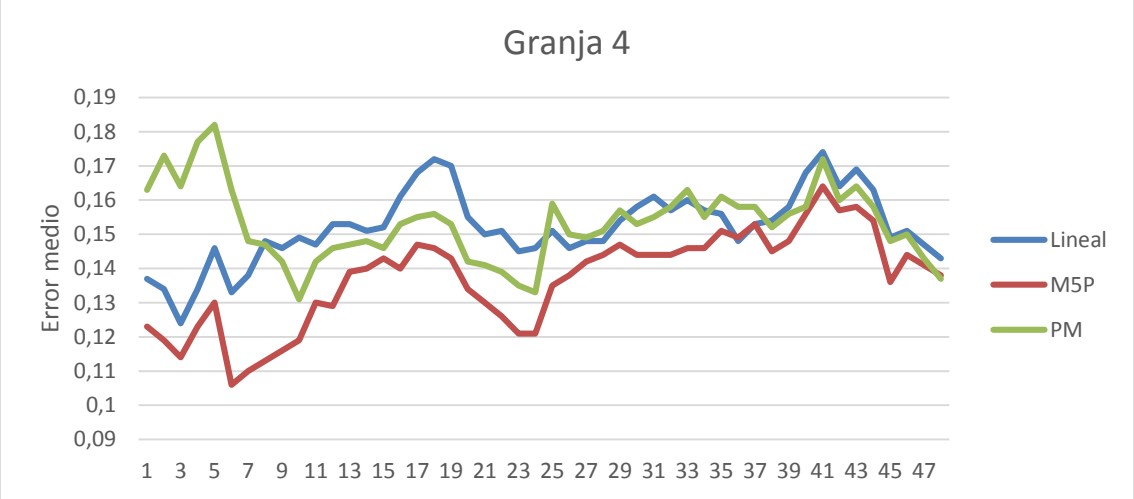
En el gráfico 17 correspondiente a la granja 2 se puede observar que el mejor método es el M5P, el Perceptron es mejor que el lineal en los primeros 24 horizontes y en los últimos es peor, en general se puede observar que el error medio de predicción para los primeros horizontes es más bajo que los últimos.

Gráfico 18. Parte III. Granja 3. Error medio de diferentes métodos para cada uno de los 48 horizontes.



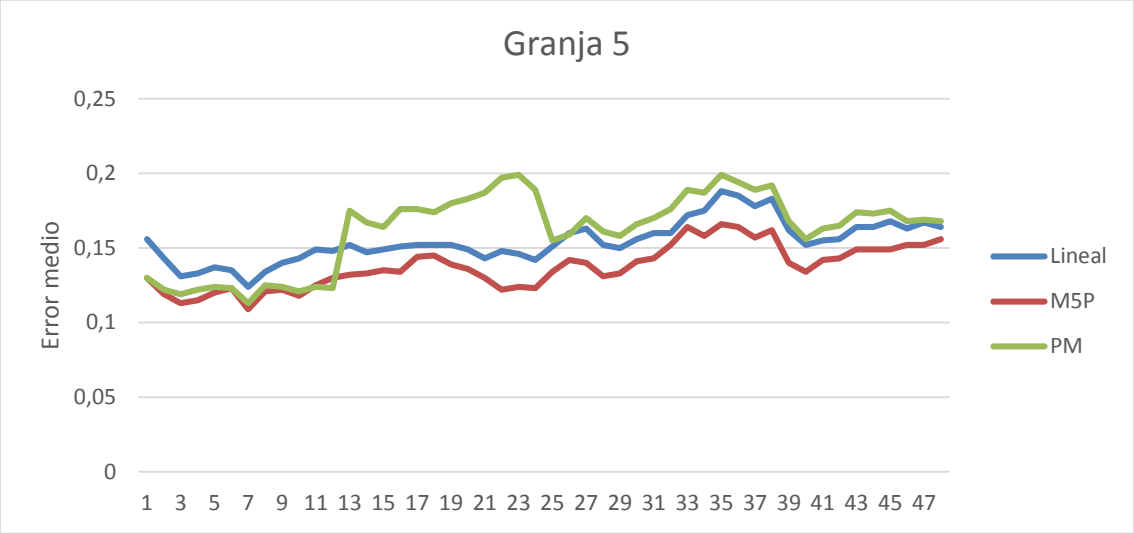
En el gráfico 18 correspondiente a la granja 3 se puede observar que el mejor método es el M5P, el Perceptron es mejor que el lineal en los primeros 24 horizontes y en los últimos es peor, en general se puede observar que el error medio de predicción para los primeros horizontes es más bajo que los últimos.

Gráfico 19. Parte III. Granja 4. Error medio de diferentes métodos para cada uno de los 48 horizontes.



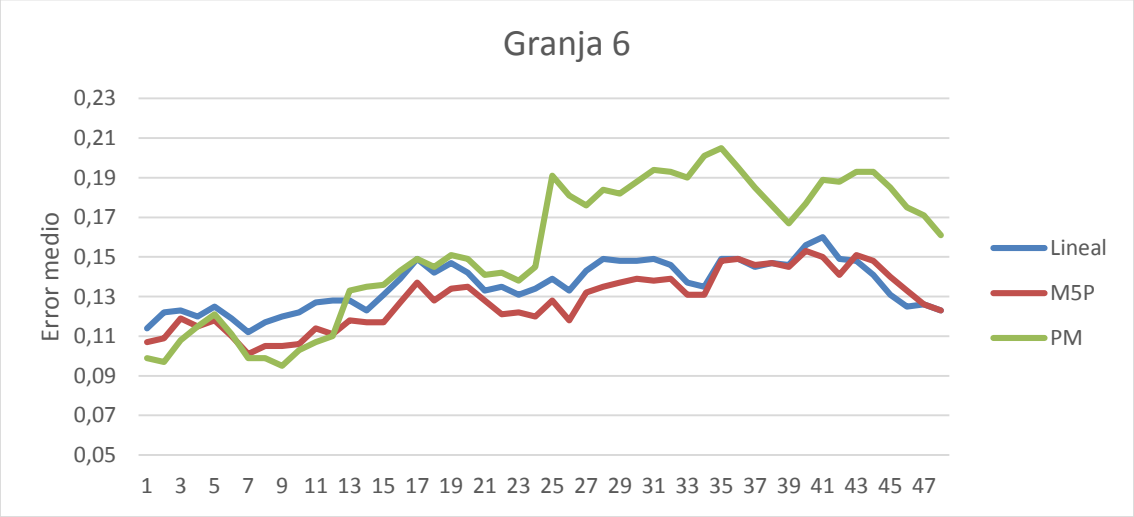
En el gráfico 19 correspondiente a la granja 4 se puede observar que el mejor método es el M5P, el Perceptron es que peor error medio ha obtenido en los primeros siete horizontes, en general se puede observar que el error medio de predicción para los primeros horizontes es más bajo que los últimos.

Gráfico 20. Parte III. Granja 5. Error medio de diferentes métodos para cada uno de los 48 horizontes.



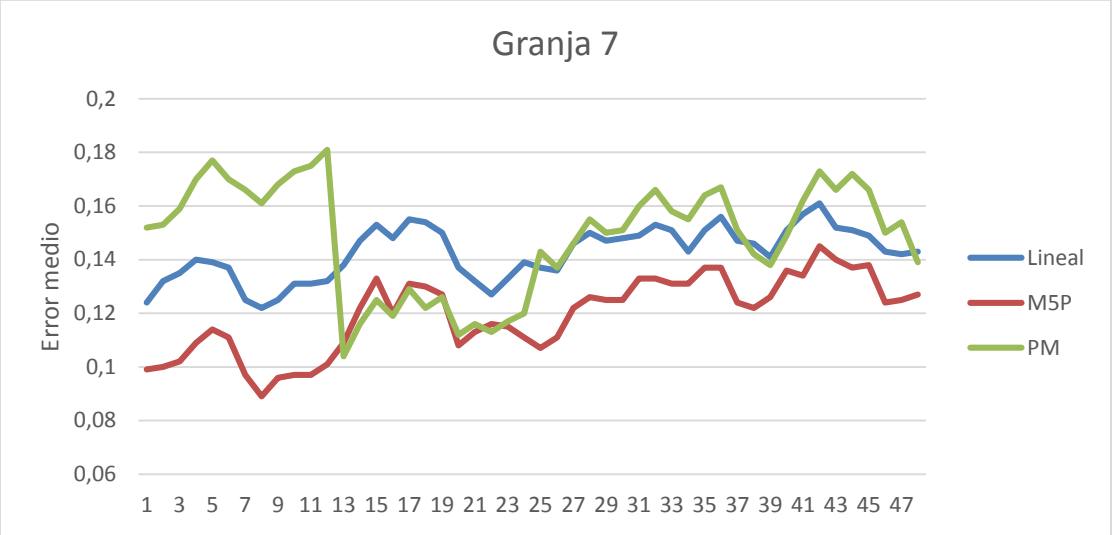
En el gráfico 20 correspondiente a la granja 5 se puede observar que el mejor método es el M5P, el Perceptron es que peor error medio ha obtenido en los últimos horizontes, en general se puede observar que el error medio de predicción para los primeros horizontes es más bajo que los últimos.

Gráfico 21. Parte III. Granja 6. Error medio de diferentes métodos para cada uno de los 48 horizontes.



En el gráfico 21 correspondiente a la granja 6 se puede observar que el mejor método en los primero 12 horizontes es el Perceptron, a partir del horizonte 13 el mejor error medio se obtiene con el M5P y el Perceptron pasa a ser el que peor resultado obtiene, en general se puede observar que el error medio de predicción para los primeros horizontes es más bajo que los últimos.

Gráfico 22. Parte III. Granja 7. Error medio de diferentes métodos para cada uno de los 48 horizontes.



En el gráfico 22 correspondiente a la granja 7 se puede observar que el mejor método en los primero 12 y los últimos 24 horizontes es el M5P y el que peor resultado obtiene es el Perceptron Multicapa.

4.3.4. Conclusiones de los resultados.

La principal conclusión del estudio realizado es que en términos generales el mejor método de aproximación es el M5P y el error medio de la predicción de la salida real para los cuarenta y ocho horizontes es más bajo en los primeros que en los últimos horizontes.

Capítulo 5. Comparación entre los resultados obtenidos en las tres partes

A continuación se muestra un resumen de los mejores resultados finales obtenidos en cada una las tres alternativas de diseño de modelos de predicción de la energía eólica, para cada una de las granjas en un horizonte de 48 horas de predicción.

5.1. Resumen de los resultados obtenidos con los diferentes métodos y modelos

En las tablas 22, 23 y 24 se muestran los mejores resultados para cada una de las aproximaciones estudiadas en este trabajo.

Parte 1.

Tabla 22. La primera alternativa, el resumen del mejor modelo de predicción evaluado con test real.

Granja Eléctrica	Método Lineal	Método PM	Método M5P
Granja1	0.1404	0.1506	0.1394
Granja2	0.1589	0.1414	0.1332
Granja3	0.1627	0.1556	0.1477
Granja4	0.1515	0.1503	0.134
Granja5	0.1529	0.1461	0.1315
Granja6	0.1353	0.173	0.1236
Granja7	0.1411	0.1282	0.1138
Error medio total	0.1490	0.1493	0.1319

Parte 2

Tabla 23. La segunda alternativa, el resumen del mejor modelo de predicción evaluado con test real.

Granja Eléctrica	Método Lineal	Método PM	Método M5P
Granja1	0.1404	0.1641	0.139
Granja2	0.1589	0.2162	0.133
Granja3	0.1627	0.1629	0.149
Granja4	0.1515	0.1394	0.1357
Granja5	0.1529	0.1425	0.133
Granja6	0.1353	0.1202	0.1272
Granja7	0.1411	0.1575	0.1149
Error medio total	0.1490	0.1575	0.1331

Parte 3

Tabla 24. Tercera alternativa, el resumen del mejor modelo de predicción evaluado con test real.

Granja Eléctrica	Media de los 3 Lineales. Test real	Media de los 3 PM. Test real	Media de los 3 M5P. Test real
Granja1	0.1405	0.148	0.1383
Granja2	0.1579	0.1552	0.1370
Granja3	0.163	0.1630	0.1525
Granja4	0.1515	0.1524	0.1367
Granja5	0.1535	0.1617	0.1367
Granja6	0.1351	0.1543	0.1276
Granja7	0.1417	0.1474	0.1187
Error medio total	0.1490	0.1546	0.1354

Viendo los errores medios obtenidos en las tres partes de este trabajo (tabla 22, 23, 24), se elige el método M5P como el mejor método de regresión, por tener el menor error (0.1319), y la mejor solución a nuestro problema de predicción de la energía eólica es la aproximación con el modelo del M5P realizada en la primera parte.

A continuación se muestra una tabla que resume los mejores resultados obtenidos en las tres partes para cada una de las granjas y aplicando el método de regresión M5P, donde incluimos también los resultados obtenidos mediante el método de la persistencia. La persistencia consiste básicamente en predecir la potencia en un instante de tiempo t con el valor de la potencia en el instante anterior, es decir Wp(t)=WP(t-1). El valor de persistencia venía dado por la competición en uno de los ficheros disponibles (concretamente en "benchmark").

Tabla 25. Resumen del mejor método de regresión, M5P con test real y la persistencia.

Granjas Eléctricas	Método M5P, test real, parte 1	Método M5P test real, parte 2	Media de los 3 M5P. Test real, parte 3	Método de la Persistencia. Test real.
Granja1	0.1394	0.139	0.1383	0.22187
Granja2	0.1332	0.133	0.1370	0.24878
Granja3	0.1477	0.149	0.1525	0.26715
Granja4	0.134	0.1357	0.1367	0.26725
Granja5	0.1315	0.133	0.1367	0.27862
Granja6	0.1236	0.1272	0.1276	0.24222
Granja7	0.1138	0.1149	0.1187	0.25527
Error medio total	0.1318	0.1331	0.1353	0.2544

Observando los resultados del M5P en las tres partes de este trabajo y los resultados obtenidos mediante el método de la persistencia (la tabla 25) se observa que para las granjas 3,4, 5, 6 y 7 se obtiene mejor resultado aplicando el M5P con el mejor modelo de la primera parte, modelo con las variables de entrada (WS,WD), mientras para la granja 1 resulta mejor el modelo con diferentes horizontes (tercera parte) y para la granja 2 resulta mejor el mejor modelo estudiado en la segunda parte con variables de entrada (HORS, WS,WD,PW(t-1)). También se observa que los resultados obtenidos en las tres partes son similares y mejores que los resultados obtenidos con el método de la persistencia.

5.2. Evolución del error medio en los 48 horizontes con el M5P en las tres partes.

A continuación compararemos cómo evoluciona el error medio obtenido en cada uno de los cuarenta y ocho horizontes y para cada una de las granjas, aplicando el mejor método de regresión M5P en las tres partes estudiadas durante este trabajo.

Gráfico 23. Granja1. Error medio del M5P para cada uno de los 48 horizontes.

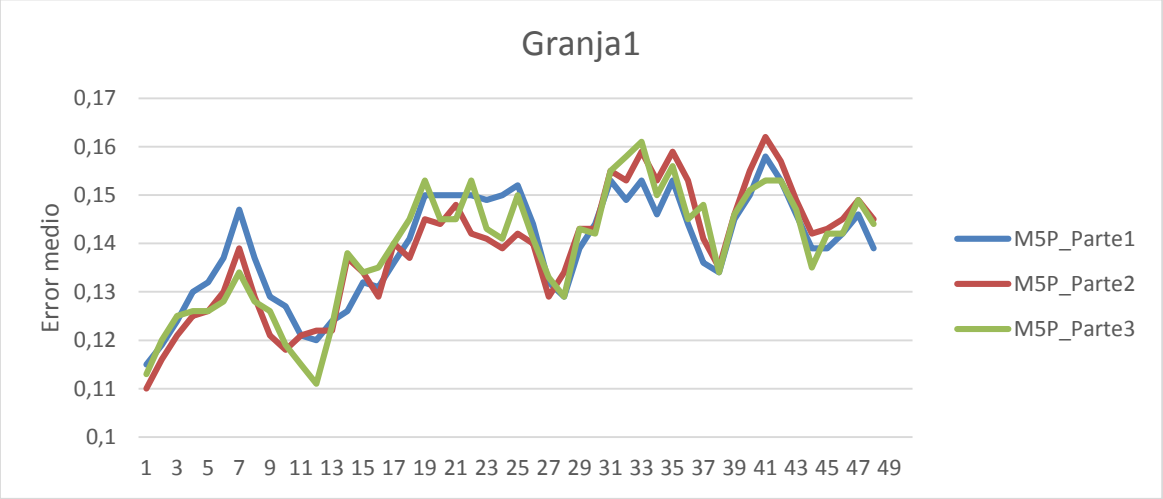


Gráfico 24. Granja2. Error medio de M5P para cada uno de los 48 horizontes.

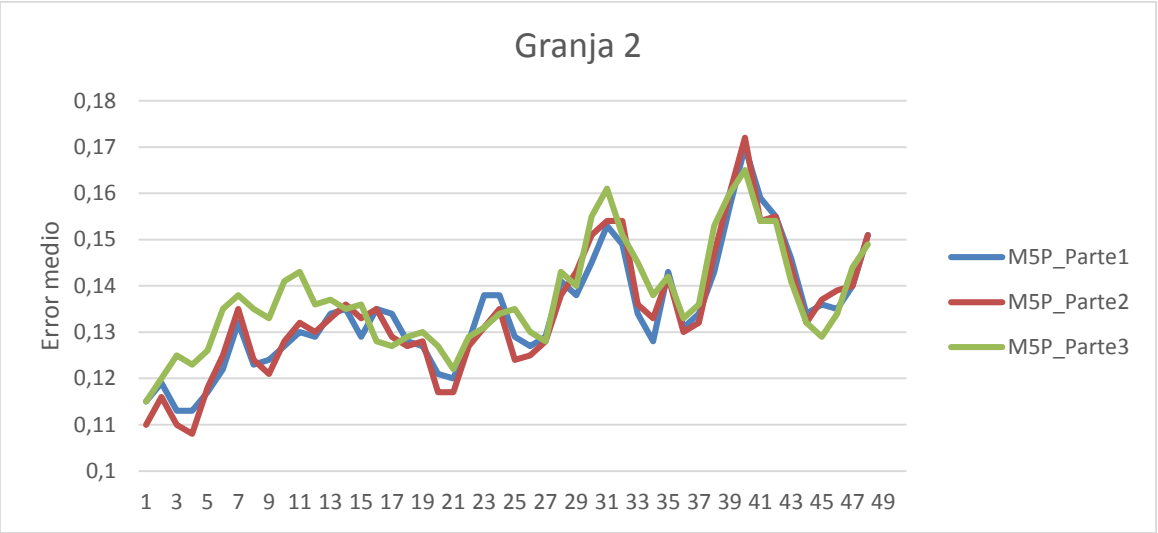


Gráfico 25. Granja3. Error medio de M5P para cada uno de los 48 horizontes.

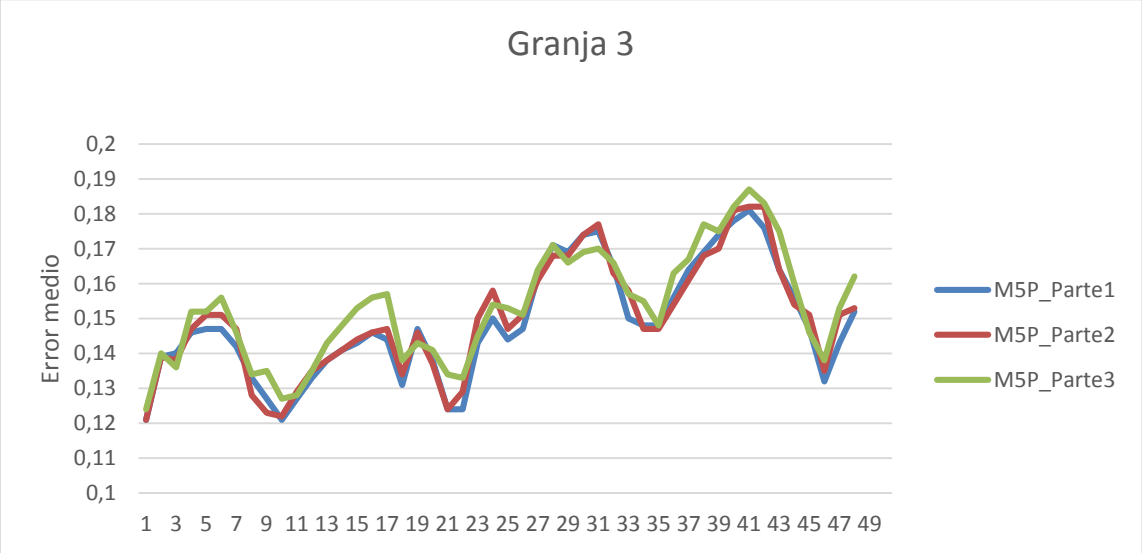


Gráfico 26. Granja 4. Error medio del M5P para cada uno de los 48 horizontes.

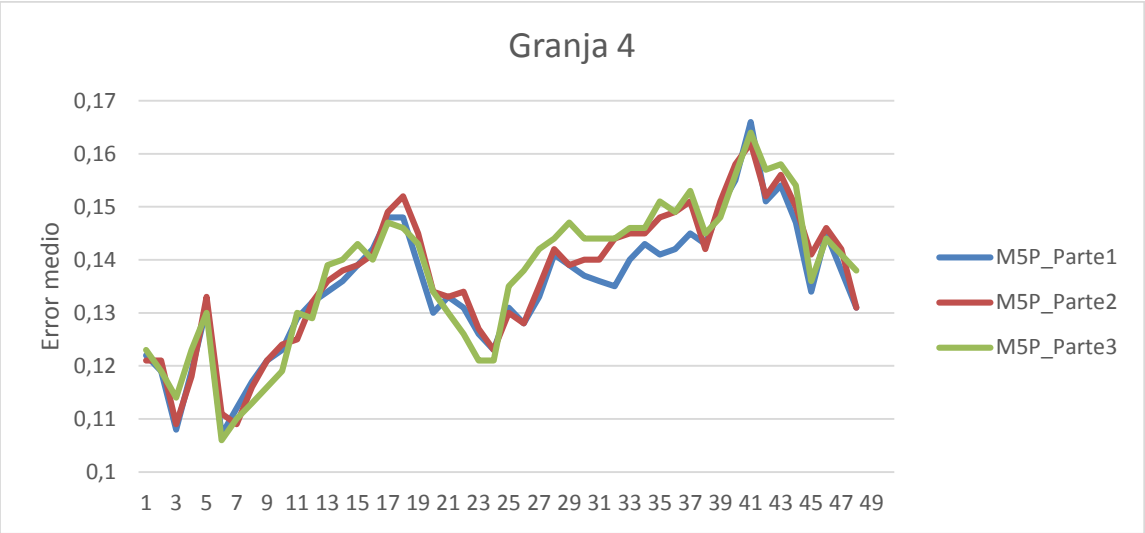


Gráfico 27. Granja 5. Error medio del M5P para cada uno de los 48 horizontes.

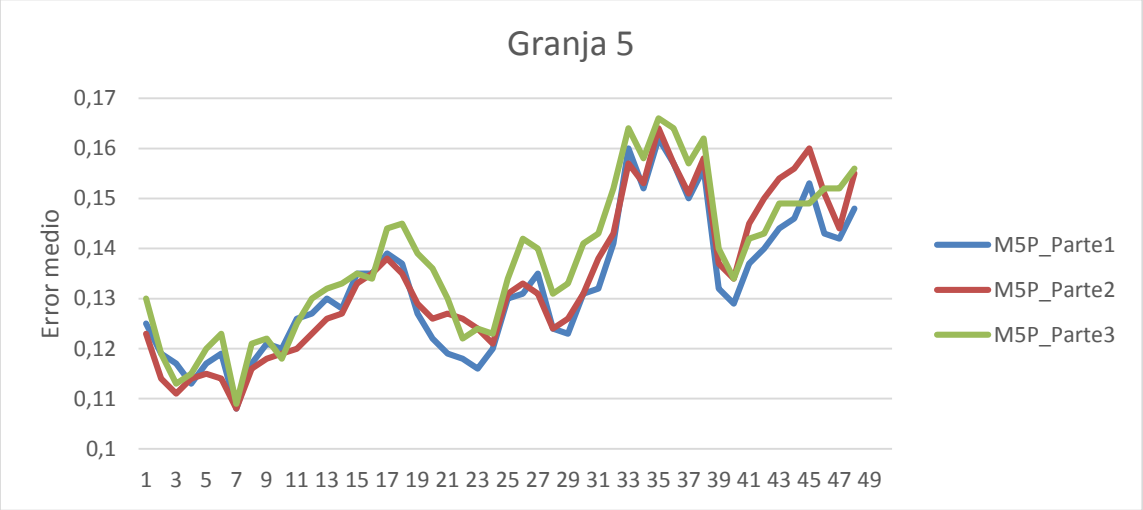


Gráfico 28. Granja 6. Error medio del M5P para cada uno de los 48 horizontes.

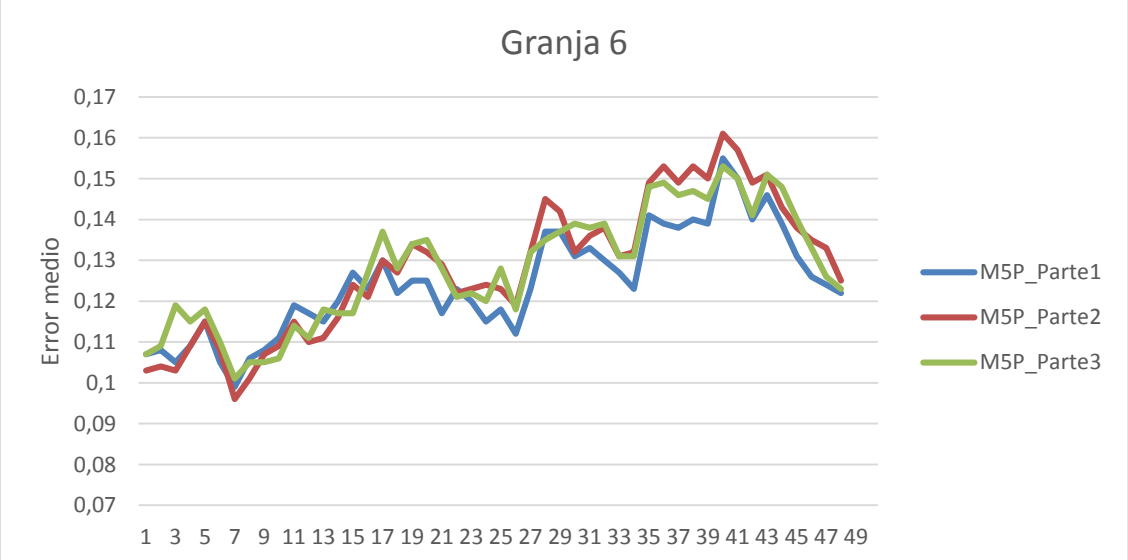
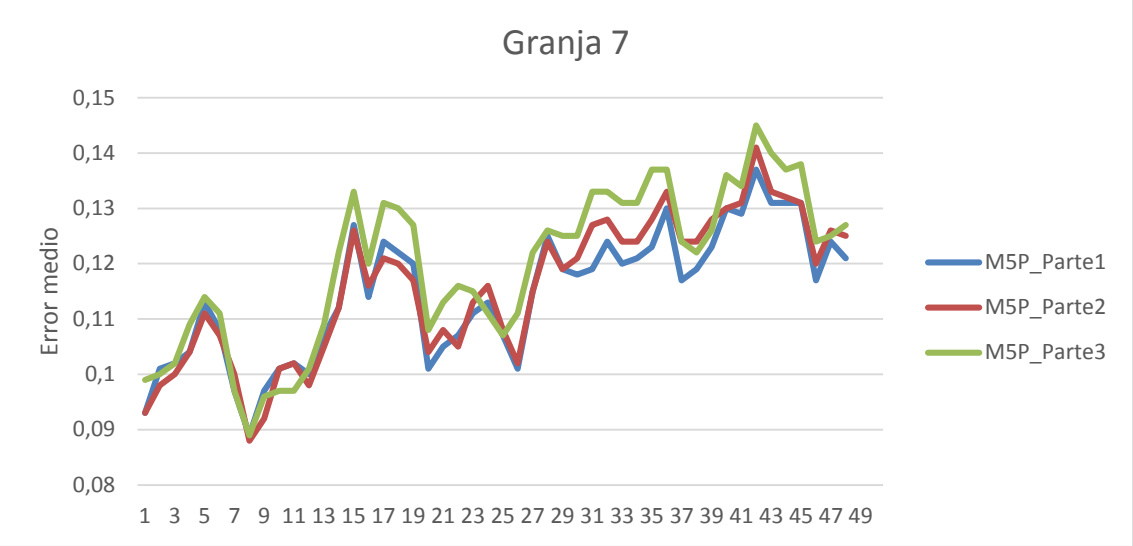


Gráfico 29. Granja 7. Error medio del M5P para cada uno de los 48 horizontes.



Analizando los gráficos para cada una de las granjas, se puede observar que el error medio aumenta y baja durante los 48 horizontes y que no hay una aproximación que sea mejor que las otras para los diferentes horizontes de predicción, de hecho no se observan de manera general grandes diferencias. Para la granja 2 y horizontes cercanos, parece que la tercera aproximación es la peor, pero este hecho no se mantiene en el resto de las granjas, por lo que no es algo concluyente. Para las granjas 5, 6, y 7 y los últimos horizontes, la primera aproximación es la que proporciona mejores resultados. La segunda aproximación tiende a ser la mejor para horizontes temporales muy cercanos (de una a seis horas).

Capítulo 6. Conclusiones y futuros trabajos

6.1. Conclusiones.

En este trabajo se han estudiado distintos algoritmos y configuraciones para predecir la energía generada en distintas granjas eólicas, a partir de predicciones meteorológicas y la serie histórica de producción. Entre los distintos algoritmos se han utilizado la regresión lineal, el perceptrón multicapa y los árboles de modelos de regresión M5P y en todos ellos se ha realizado una exploración de posibles valores para sus parámetros. El trabajo tiene tres partes, en el primero se estudia qué variables de predicción meteorológica son relevantes, en el segundo si añadir instantes anteriores de la serie histórica de producción mejora la predicción. Y en la tercera, si se pueden mejorar las predicciones descomponiendo el modelo global en distintos modelos especializados en horizontes a corto plazo (hasta 12 horas), medio (hasta 24 horas) y largo plazo (hasta 48 horas).

Después de finalizar todos los estudios y análisis planificados para este trabajo se ha comprobado que las únicas variables atmosféricas de interés para la predicción de producción de la energía eléctrica en los siete parques eólicos son, esencialmente, la velocidad y la dirección del viento (WS Y WD).

Tras realizar la comparación entre los mejores resultados obtenidos en las tres partes de estudio y análisis durante este trabajo, hemos llegado a la conclusión de que el mejor método de regresión para realizar la predicción de la energía eólicas en cada uno de los parques eólicos es el método basado en árboles de modelos de regresión M5P.

La creación de modelos especializados (tercera parte del trabajo) para tres grupos de horizontes temporales no mejora los resultados de manera global, pero tampoco para ninguno de los tres grupos de horizontes. Sin embargo, usar las variables meteorológicas junto a un instante de la serie histórica de producción (segunda parte del proyecto) sí que mejora las predicciones a muy corto plazo (menos de 6 horas), aunque la diferencia no es grande con el modelo que no usa instantes anteriores de la serie. Teniendo en cuenta los resultados en media de todos los horizontes temporales en conjunto (de 1 a 48 horas), y como conclusión principal del proyecto, el mejor modelo es el que usa dos (WS, WD) de las cuatro variables disponibles con el método M5P y sin ningún instante de la serie temporal (aunque usando un instante los resultados son muy similares).

Por último se ha confirmado que a medida que aumenta el horizonte temporal tiende a aumentar el error de predicción.

6.2. Futuros trabajos.

Estudiar y analizar métodos predictivos de la energía eólica como métodos de regresión mediante máquinas de vectores de soporte, ya que en este caso de estudio no ha sido posible probar este método de regresión debido a que el tiempo de generación de modelos es muy alto (tarda entre 30 hasta 60 minutos el proceso de entrenamiento con los datos del problema actuales).

También se puede profundizar en el estudio y análisis de métodos de regresión para construir modelo predictivos a muy corto plazo 6 horas, (modelos con serie temporal, con serie anterior de variable de predicción o con serie anterior de la producción).

Por último se puede continuar este trabajo desarrollando una aplicación de escritorio en el lenguaje de programación Java que realiza la predicción de la producción de la energía o potencia eléctrica de origen eólico a corto plazo o a 48 horas.

Para poder desarrollar esta aplicación se puede utilizar la librería java que ofrece Weka, esta librería proporciona todas las clases y métodos para la creación y evaluación de los modelos de predicción.

Esta aplicación debe permitir al usuario crear nuevos modelos o usar modelos existentes (modelos creados anteriormente y guardados en el disco duro), para ello la aplicación debe ser capaz de cargar los datos de entrada correspondientes a los parques eólicos correspondientes, transformar y generar los conjuntos de entrenamiento (generar, entrenar los datos con los mejores métodos de regresión estudiados anteriormente), una vez que la aplicación crea los mejores modelos, debe permitir al usuario mediante una interfaz introducir los valores de los atributos meteorológicos actuales e iniciar la predicción, cuando el usuario inicia la predicción, la aplicación debe mostrar los resultados de la predicción para los 48 horas.

Capítulo 7. Presupuesto y planificación

7.1. Planificación

En general las principales tareas que existen para la planificación son las siguientes:

Recopilación y búsqueda de información: Buscar información sobre la energía eólica, y los modelos más actuales de predicción de la energía eólica, su entorno socio económico, etc.

Búsqueda de la información y planificación de la solución: Buscar información acerca de las herramientas de análisis de datos más utilizadas y así como las técnicas de aprendizaje automático más usadas para resolver problemas de aprendizaje supervisado.

Diseño de modelos de predicción: En esta fase se ha ido diseñando las tres soluciones de forma secuencial, se ha realizado la implementación de diferentes funciones para tratar los ficheros de datos de entrada y salida y generar las diferentes estructuras del conjunto de entrenamiento y test para generar los diferentes modelos de predicción.

Validación de modelos de predicción: En esta fase se ha realizado el estudio de los mejores parámetros para los diferentes métodos de regresión, así como validar los diferentes modelos de predicción generados.

Evaluación de modelos de predicción: En esta fase se ha realizado la evaluación de los mejores modelos de predicción.

Documentación: Redacción del documento a entregar. Realmente este documento se ha ido redactando incrementalmente a medida que se completaba una tarea, para que no se olvide ningún detalle.

A continuación se muestra mediante una tabla con todas las tareas llevadas a cabo e iteración o ciclo en las que se han realizado dichas tareas, para alcanzar todos los objetivos de este trabajo de forma satisfactoria.

Tabla 26. El resumen de la planificación.

Actividad Principal	Ciclo	Fecha de inicio	Duración (días)	Fecha de fin
Recopilación de información	1	15/09/2014	6	22/09/2014
Búsqueda de la información y planificación de la solución	2	23/09/2014	10	06/10/2014
Generar el conjunto de entrenamiento y test, primera alternativa, (parte 1).	3	07/10/2014	6	14/10/2014
Realizar el estudio de parámetros y validar los modelos (parte 1)	4	15/10/2014	11	29/10/2014
Evaluar el mejor modelo (Parte 1).	5	30/10/2014	6	06/11/2014
Generar el conjunto de entrenamiento y test, segunda alternativa, (parte 2).	6	07/11/2014	11	21/11/2014
Realizar el estudio de parámetros y validar los modelos, (parte 2)	7	24/11/2014	11	09/12/2014
Evaluar el mejor modelo (Parte 2).	8	10/12/2014	6	17/12/2014
Generar el conjunto de entrenamiento y test, tercera alternativa, (parte 3).	9	18/12/2014	10	02/01/2015
Evaluar los modelos, (Parte 3).	10	05/01/2015	6	13/01/2015
Documentación	11	14/01/2015	23	16/02/2015

En la tabla 26 se puede observar la fecha de inicio, la fecha de fin, el número de días trabajados y el ciclo en el que se realizó cada una de las actividades necesarias para alcanzar los objetivos de este trabajo, teniendo en cuenta que el número de horas trabajadas por día es de 3 horas.

En general todas las actividades se han llevado a cabo con éxito, ya que el tiempo estimado para cada una de ellas se ha finalizado de forma satisfactoria.

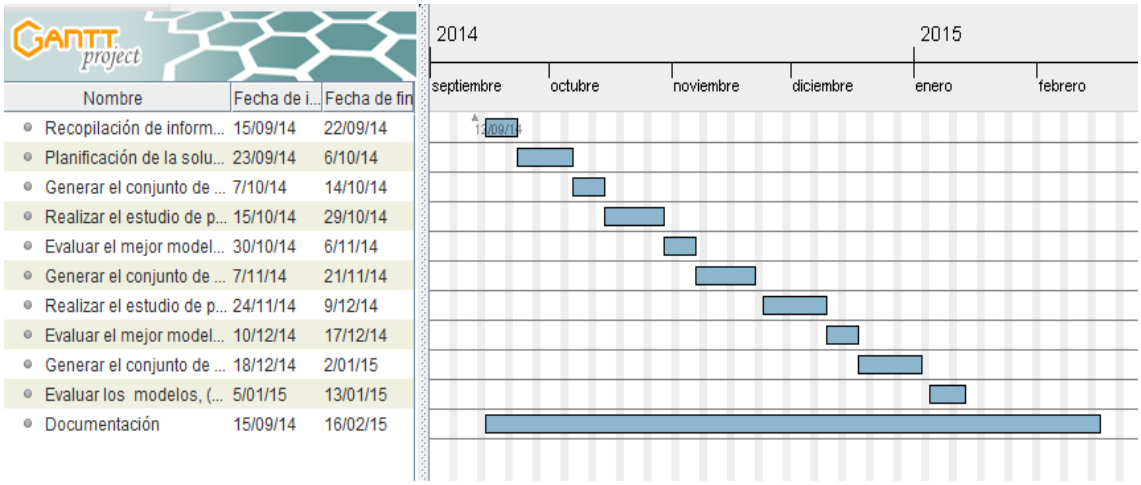
En cuanto a la documentación, aunque hay solo un ciclo en el que se dedica a la documentación, se empezó a documentar desde el ciclo 1 puesto que es necesario para no olvidar ningún detalle.

Al final de cada ciclo se han ido realizando reuniones de seguimiento, en este caso, con el tutor del proyecto, para comprobar el estado final de dicho ciclo, además de asignar los nuevos objetivos del siguiente ciclo.

Por último, añadir que si en algún momento de algún ciclo ha sido necesario modificar alguna información del ciclo anterior, se ha realizado sin ningún problema.

A continuación se muestra el diagrama de Gantt que muestra la planificación.

Gráfico 30. Diagrama de Gantt con la planificación.



7.2. Presupuesto

A continuación se especifica el presupuesto estimado para el proyecto, especificando los gastos totales de este trabajo que se componen de los gastos del personal, hardware y software que han sido necesarios para la realización del proyecto.

Para la realización del proyecto ha sido necesaria la compra de un ordenador portátil con las siguientes características:

Acer E5-571G-51WG NX.MSCEB.001

- Procesador: Intel Core i5-5200U a 2.2GHz
- Memoria RAM 4GB DDR3
- Disco duro 1TB SATA 5400rpm
- SO: Windows 8.1 64-bit

Precio IVA incluido: 546,00 €.

Para la realización del proyecto no es necesaria la compra de ningún producto software (excepto el Microsoft Office) ya que se van a usar aplicaciones de licencias libre (gratuita).

Tabla 27. Coste por software.

Software	Precio de licencia
Windows 8.1	Incluido en el precio del portátil
Microsoft Office 2013	10,00 € /mes, incluido IVA.
Eclipse (versión 4.3.0, para Windows)	0 €
Weka (versión 3.7.12.0)	0 €

Tabla 28. Coste total por software.

Software	Duración en meses	Coste por mes	Uso dedicado	Coste total
Microsoft Office 2013	5	10,00 €	100	50,00 €

Para estimar los costes del personal se tiene en cuenta la planificación realizada anteriormente y el número de personas que realizan este trabajo, en este caso realiza el trabajo solo una persona, a continuación se muestra una tabla con el coste total del personal.

Tabla 29. Coste total por el personal.

Total días trabajados	Horas trabajadas al día	Coste por hora	Coste total
106	3	20 (IVA incluido)	6360,00 €

A continuación se presenta el presupuesto total.

Tabla 30. Presupuesto total.

Concepto	Coste
Hardware	546,00 €
Software	50,00 €
Personal	6360,00 €
Total IVA incluida.	6956,00 €

Capítulo 8. Referencias

A continuación se muestra las principales fuentes bibliográficas que han sido consultadas para la realización de este trabajo de fin de grado:

[1] Foley, A. M., Leahy, P. G., Marvuglia, A., & McKeogh, E. J. (2012). Current methods and advances in forecasting of wind power generation. Renewable Energy, 37(1), 1-8.

[2] Hong, T., Pinson, P., & Fan, S. (2014). Global energy forecasting competition 2012. International Journal of Forecasting, 30(2), 357-363.

[3] Markus Hofmann, Ralf Klinkenberg. RapidMiner: Data Mining Use Cases and Business Analytics Applications Chapman & Hall/CRC (2013).

[4] R Core Team (2013). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL http://www.R-project.org/.

[5] Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. (2009). The WEKA data mining software: an update. ACM SIGKDD explorations newsletter, 11(1),10-18.

[6] [7] Pedro Isasi e Inés M. Galván. Redes de Neuronas Artificiales: Un enfoque práctico. Pearson. Prentice Hall. Madrid 2004.

[8] Ross J. Quinlan: Learning with Continuous Classes. In: 5th Australian Joint Conference on Artificial Intelligence, Singapore, 343-348, 1992.

[9] Witten, I. H., Frank, E., Trigg, L. E., Hall, M. A., Holmes, G., & Cunningham, S. J. (1999). Weka: Practical machine learning tools and techniques with Java implementations.

Libros recomendados para realizar este trabajo.

Pedro Isasi e Inés M. Galván. Redes de Neuronas Artificiales: Un enfoque práctico… Pearson. Prentice Hall. Madrid 2004.

Aprendizaje automático. Conceptos básicos y avanzados: Aspectos prácticos utilizando el software Weka. Sierra Araujo, Basilio. Pearson/Prentice Hall, 2006.

El libro de aprendizaje automático de Pedro Isasi Viñuela y Daniel Borrajo Millan, editorial Sanz y Torres, 2006.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. (2009). The WEKA data mining software: an update. ACM SIGKDD explorations newsletter, 11(1), 10-18.